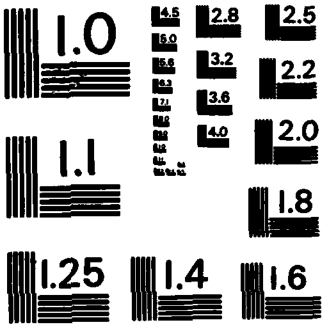
AD-A121 793   ADVANCED AVIONIC SYSTEMS FOR MULTIMISSION APPLICATIONS   1/1
VOLUME I(U) BOEING MILITARY AIRPLANE CO SEATTLE WA
L A SMITH ET AL. OCT 82 AFWAL-TR-82-1076-VOL-1
UNCLASSIFIED   F33615-77-C-1252   F/G 9/2   NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AFWAL-TR-82-1076
Volume I

ADVANCED AVIONIC SYSTEMS FOR MULTIMISSION APPLICATIONS

Boeing Military Airplane Company
Seattle, Washington 98124

October 1982

Final Report for Period January 1978 - June 1980

Approved for public release, distribution unlimited.

AVIONICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Claude M Fletcher, Jr

CLAUDE M FLETCHER, JR
PROJECT ENGINEER
Mission Software & System
Integration Group

L. DANIEL SNYDER, Chief
Mission Software & System
Integration Group
System Avionics Division

FOR THE COMMANDER

Frank Scarpino

FRANK A SCARPINO
Acting Chief, System Avionics Division
Avionics Laboratory

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/AAAS-1, *W-PAFB, OH 45433 to help us maintain a current mailing list".*

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFWAL-TR-82-1076, Volume I	2. GOVT ACCESSION NO. AD-A121793	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) ADVANCED AVIONIC SYSTEMS FOR MULTIMISSION APPLICATIONS		5. TYPE OF REPORT & PERIOD COVERED Final Report for Period January 1978 - June 1980
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Leroy A. Smith, W. Al Crossgrove, Donald E. Dewey et al.		8. CONTRACT OR GRANT NUMBER(s) F33615-77-C-1252
9. PERFORMING ORGANIZATION NAME AND ADDRESS Boeing Military Airplane Company Seattle, Washington 98124		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project 2003, Task 01, Work Unit 10
11. CONTROLLING OFFICE NAME AND ADDRESS Avionics Laboratory (AFWAL/AAAS-1) Air Force Wright Aeronautical Laboratories (AFSC) Wright-Patterson Air Force Base, Ohio 45433		12. REPORT DATE October 1982
		13. NUMBER OF PAGES 49
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Presented at 1980 NAECON conference as "Advanced Avionic Systems for Multimission Applications" by L. A. Smith and D. E. Dewey.
The computer data contained in this technical report are theoretical and in no way reflect Air Force-owned software programs.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computers, avionics, microprocessors, technology forecast, hardware standards, software standards, information transfer systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This study specifically addressed the role of microprocessors and LSI technology (five year forecast) in future avionic systems data bus, information transfer system designs which allow future avionic systems additions, and the impact of Air Force standards on microprocessors. Cost models for hardware and software and information transfer system simulation models were also reviewed for possible use in the analysis of these systems. The study has resulted in the definition and analysis of three data bus information transfer systems which allow multimission applications. These information transfer systems are called

(Cont'd)

Block 20

stationary master, nonstationary master and contention multiple access.

FOREWORD

VOLUME I

This final technical report for the Advanced Avionic Systems for Multi-Mission Applications (AASMMA) was prepared by The Boeing Military Airplane Co. (BMAC), Seattle, Washington. The final report consist of three separately bound volumes which covers the work performed under contract F33615-77-C-1252 during the period of January 1978 to June 1981.

The program was performed in two phases for the Air Force Wright Aeronautical Laboratories, Wright-Patterson AFB, Ohio 45433. The first phase covered three tasks which addressed (1) Distributed Avionics Information System Design, (2) Avionic Cost Analysis Methods & Models, and (3) Embedded Microcomputer Standardization Concepts. These tasks were conducted for AFWAL/AAAA. The contract monitor was Mr Gary Wambold, the program manager was Mr Donald E Dewey, and the principal investigators were Dr Leroy A Smith and Mr Al Crossgrove. Volume I of this report describes this phase.

The second phase of the program Volumes II and III covered tasks which addressed (1) the Development & Evaluation of Advanced Digital Avionics System Architectures and (2) the Development of a Single Processor Synchronous Executive (SPSE) derived from the Digital Avionics Information System (DAIS) Executive. These tasks were conducted for AFWAL/AAAS and the AFWAL contract manager was Mr Claude M Fletcher, Jr, the Boeing program manager was Dr Leroy A Smith, and the principal investigator was Mr Stephen W Behnen.

The other significant contributers to this effort included:

Richard F Bousley
Tammy R Cremeen
Dr Robert L Gutmann
John J Henrick
James H Mason
Mack B McCall
Kevin M McMahon
Michael E McSharry
Keith D Pratt
Gerald Sommerman
Laura Townsend
Frank E Troth
C Ray Turner

all from The Boeing Company; James Gracia, Edward Comer, and Joseph Malnar, all from the Harris Corporation; Capt Robert Percefull from the U S Air Force; and Lynn Trainor from the Systran Corporation.

TABLE OF CONTENTS

TABLE OF CONTENTS (Concluded)

LIST OF ILLUSTRATIONS

LIST OF TABLES

1.0 INTRODUCTION

1.1 SCOPE

This technical report summarizes the activities and results of the Advanced Avionic Systems for Multimission Applications (AASMMA) program. Phase 1 of the AASMMA program was to study current and projected information transfer system designs and architectures for avionics systems which require a multimission capability. The purpose and background of this phase are summarized in paragraphs 1.2 and 1.3. Paragraph 2.0 describes the program tasks. Paragraph 3.0 summarizes the study results. Paragraph 4.0 expands on the study results. Most of the content of this volume is a summary of the detailed work documented in the first AASMMA Interim Report, Volumes 1 and 2, and Appendices A through L. Phase 2 of the AASMMA program is summarized in Volume 2. The second phase examined and documented in depth the information transfer systems defined in Phase 1. A compact version of the DAIS executive which functions in a one processor system and supports only synchronous bus communications was designed, developed, and delivered.

1.2 BACKGROUND

In the past few years it has been the goal of the Air Force to develop and apply methods and technologies that would permit avionic systems to evolve in an orderly manner as mission needs change. A lack of any type of interface commonality among avionic systems has made the task of system design and integration, as well as the task of upgrading or modifying systems, very costly. The Digital Avionics Information System (DAIS) concept was established by the Air Force to investigate and establish standard interfaces among the various elements of avionic systems to simplify and reduce this cost. These concepts are now mature and are being considered for near-term retrofit airplane programs and are planned for future systems.

The concept of multimission roles for a single airframe (or a restricted family of airframes) is influencing our military weapon planning. Threats, which are changing more rapidly than ever before, make it necessary to plan for mission-adaptive and threat-adaptive avionic suites over the life of an airframe. Two multimission concepts are emerging. One approach is to design a "core" set of avionics and separable "peripheral" avionics so that the avionics suite can be readily changed by removing and replacing mission dependent functions. Another approach is to depend on well established interface standards (e.g. standard hardware and software modules) which permit an avionic system to be updated (retrofit) throughout the life of the airframe. These approaches are not mutually exclusive, and they can be complementary. Therefore, what is presently required is the adaptation of current interface standards and the exploitation of new digital technologies to achieve the multimission capability.

Multimission roles can be readily accomplished if the multimission functions can be isolated and made independent. Isolation (independence) between functions can be achieved by using the inherent separation of functions found in hierarchical architectures. Such architectural features would make it easier to develop, integrate, maintain and modify (update) an avionics system or to use it in an aircraft having multimission roles. Most avionic technologies and design standards, including the DAIS related standards, were designed primarily for single level architectures using minicomputers as the processing elements. However, multilevel or hierarchical architectures require a high degree of distributed processing. This factor has discouraged the use of this architecture because of the high cost of military minicomputers. Now that microprocessor technology has progressed to the point where military qualified microcomputers cost substantially less than military qualified minicomputers, the hierarchical architecture using distributed processing can become a reality.

1.3 PURPOSE

The AASMMA program had two principal concurrent purposes. One purpose was to perform a study of future avionic systems. The other purpose was to modify the existing DAIS executive so that it can be more readily applied to single processor system integrations. Certain retrofit programs and future avionic programs that have limited integration requirements can use this kind of single processor executive.

The central issues that were addressed in the AASMMA study program were: (1) Is the hierarchical, distributed processing architecture a cost-effective means of meeting the needs of multimission avionic systems in the future? (2) If so, then what new technologies are required and what design guides must be followed to insure an effective avionic system architecture development? (3) What changes should be made to the DAIS concept of interface standards and processing element designs so it can be used effectively in hierarchical, distributed processing architectures oriented toward multimission requirements? (4) Can the DAIS executive program size be reduced substantially by utilizing only synchronous bus traffic and by using a single processor to perform the applications programs? (5) Can this new executive design be made expandable to include multiple synchronous processors with minimal penalty?

One goal of the AASMMA program was to maintain compatibility with as many of the existing avionic system designs and standards as possible. Another goal was to develop designs and concepts which recognize the real-world constraints of: (1) avionic integrator and subsystem supplier roles, and (2) the limited interest in military products by commercial electronic suppliers due to low volume production requirements and exacting performance standards.

2.0 AASMMA Program Summary - ~asks I, II, and III

Phase 1 of the AASMMA program consisted of Tasks I, II, and III. Task I was a preliminary design and evaluation of candidate Information Transfer Systems (ITS). This task also identified the changes to DAIS standards to suppor: the new ITS and included a technology forecast. Task II developed cost models for evaluating avionic configurations and standards. Task III was a review, evaluation and selection of various software and hardware standards. Figure 2-1 shows the master program schedule for AASMMA.

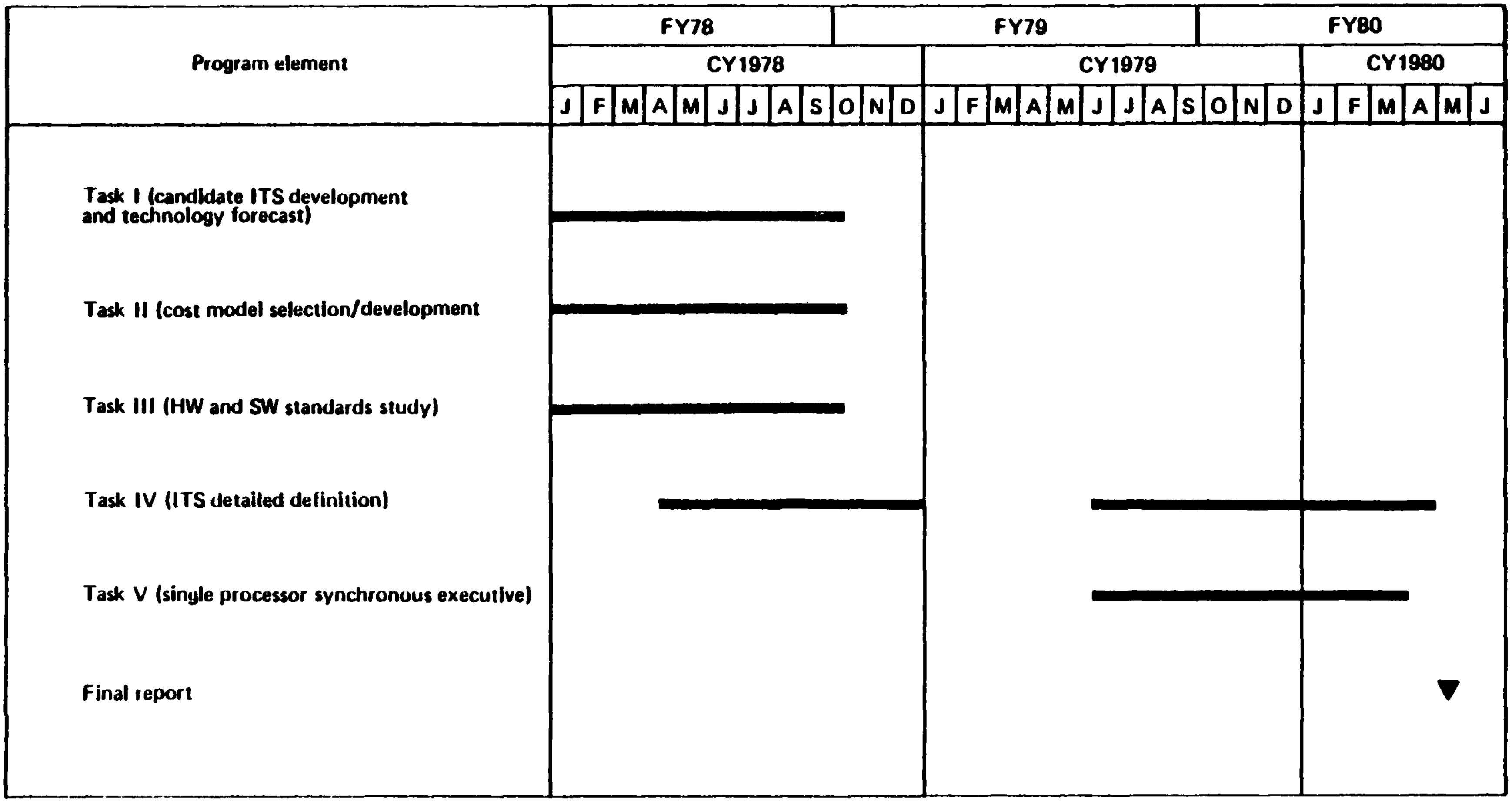


Figure 2-1. Master Program Schedule

3.0 SUMMARY OF TASK FINDINGS

Two significant, general observations were made about the multimission requirements and the microprocessor based technology:

1) To meet the requirements of a multimission avionic system, the changing components within the avionic system to meet the multimission requirements must be functionally independent. Functional independence in avionic design is achieved in three ways: (a) use of an Information Transfer System (ITS) that permits the addition or deletion of functions and equipment without impacting the operation of other functions, (b) use of a system network approach that achieves independent operation between the various levels of the physical and logical structure and (c) use of functionally independent software modules. All three methods are discussed later in this report.

2) The impact of microprocessor technology on avionic integration methods will be minimal. The introduction of microprocessors into sensors and avionic integrating elements is not going to reopen the issues of HOL versus AL, software development methods, or specialized versus general purpose processors. This is because: (a) microprocessors available in 1982 will be equivalent to the minicomputers of 1978 (when this study began) and (b) a standard approach to both operational and support software will be carried through to microprocessors. The impact of the microprocessor technology on future avionics integration methods is apparent. For the first time, multilevel, hierarchical processing networks with distributed processing elements will be feasible. This provides a method to meet the multimission requirement described in (1) above. With this new technology comes the need to change existing ITS designs and establish new guidelines in the application of existing standards. Some of the changes required to one ITS design (DAIS) are reported in Paragraph 4.2.

3.1 TASK I. SUMMARY

3.1.1 Candidate Information Transfer Systems (ITS)

Three candidate information transfer systems (ITS) that can meet the requirements of multimission and hierarchical bus structures were defined:

1) the stationary master ITS is a DAIS-like ITS in which a single master controls all the communications for a single bus pair.

2) the nonstationary master ITS, as originally derived in Task I, is a polled ITS in which each potential master responds to a polling request with a message priority. Once the active master has established which potential master has the highest message priority, control is transferred to the new master. Then the new master operates as a stationary master until its transmission interval is complete. In Task IV, the nonstationary bus control transfer scheme was modified to a round robin

protocol to improve efficiency. In the round robin scheme each master transfers control to a predefined new master.

3) the contention access protocol, the third approach, is a completely different ITS in which each device contends for master bus control. Once a device has obtained control, it transmits messages until its transmission interval is completed.

The goal of each ITS design was to implement a core avionic system which could remain fixed while mission equipment could be added or deleted without impacting the core system. Both the nonstationary and contention access ITS meet this goal. Each nonstationary master has the capability to control specific functions in the multimission system. One master controls the core avionics and one or more masters control the additional devices which use data extracted from the core system. Contention access processors operate on broadcast data when the data becomes available or when it is requested. Additional mission equipment and functions can be added to the architecture causing them to contend with the core avionic equipments for bus usage. The stationary master ITS can not meet the goal of a fixed core avionic system because the executive data base must be modified in the master for each change. The alterations to the master may be highly undesirable because no elements of the core avionics have been changed. For example, if a flight control processor were used as a master, each change to the remainder of the avionic system could cause some measure of revalidation to be required.

High priority asynchronous message timing was thought to be a potential problem with both nonstationary and contention, but analysis has shown that the contention access ITS has a better average asynchronous message response time than the stationary master, while the nonstationary master is slightly worse. Figure 3-1 shows the average wait time for an asynchronous message. Bus efficiency was also thought to be a potential problem with the nonstationary and contention approaches. The contention ITS was of special concern because as traffic becomes very heavy, the potential number of message collisions increases. However, with the proper selection of design parameters the contention system can operate more efficiently than the other ITS (figure 3-2).

Analysis of the nonstationary master ITS indicated that it has more overhead than the stationary master ITS for bus communication, as can be seen in figures 3-1 and 3-2. Therefore, the only reason to move from a stationary master to a nonstationary master concept is to provide the multimission capability. If no mission change is ever contemplated for a system, the stationary master is the best choice.

The contention master has the advantages offered by ARINC commercial data bus standards in that it provides: 1) independent bus controllers for each device rather than a select number of controllers for a bus and 2) a data format which identifies messages by type rather than by device. An additional advantage of the contention master design over the ARINC design is the common bus structure which facilitates the integration of data and functions. The contention system can also operate in a synchronous or asynchronous manner providing the maximum flexibility to meet multimission requirements of retrofit and short term mission reconfiguration. The major difficulty associated with contention is its incompatibility with MIL-STD-1553.

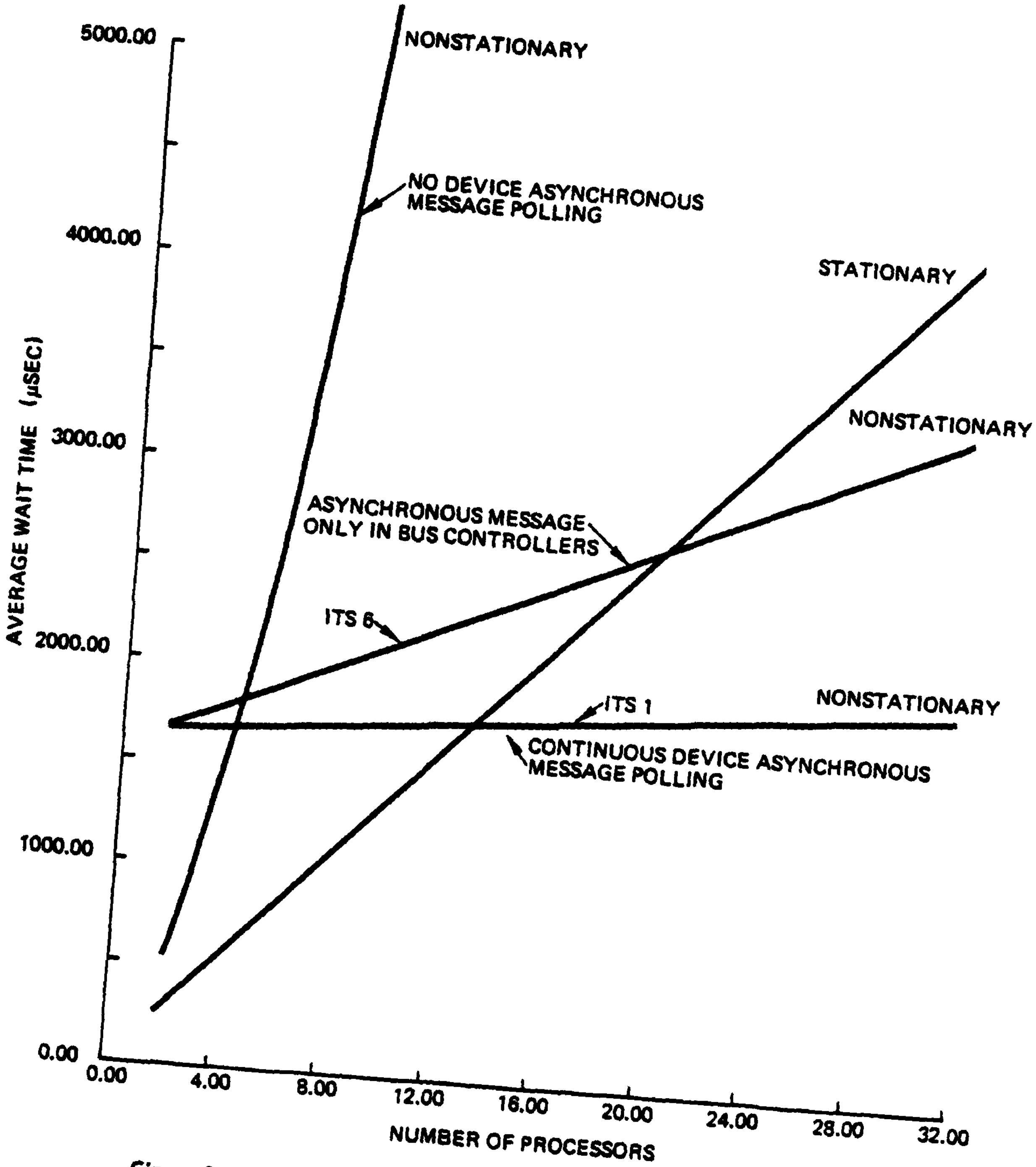
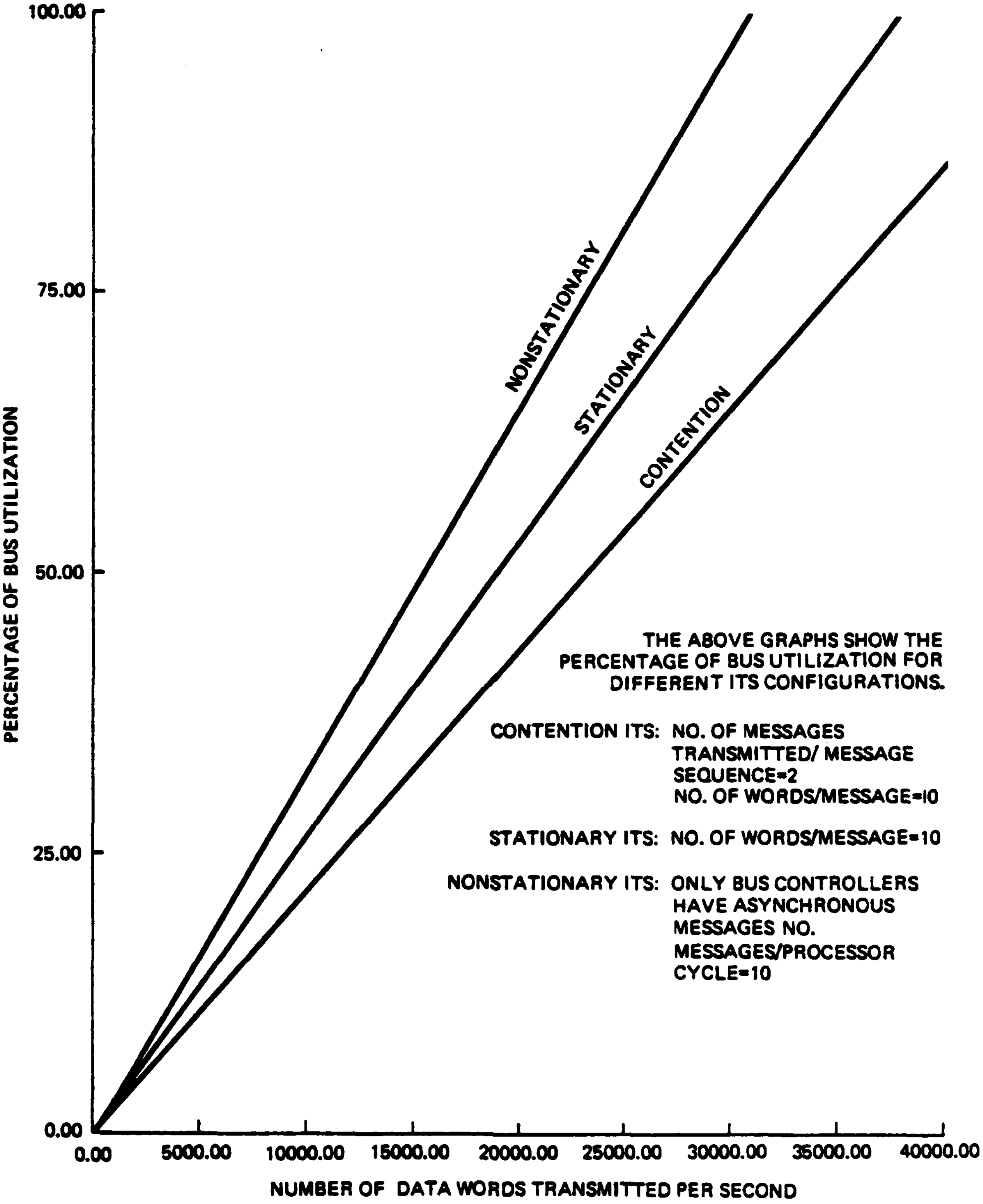


Figure 3-1. Average Wait Time For High Priority Asynchronous Messages



THE ABOVE GRAPHS SHOW THE PERCENTAGE OF BUS UTILIZATION FOR DIFFERENT ITS CONFIGURATIONS.

CONTENTION ITS: NO. OF MESSAGES TRANSMITTED/ MESSAGE SEQUENCE=2
NO. OF WORDS/MESSAGE=10

STATIONARY ITS: NO. OF WORDS/MESSAGE=10

NONSTATIONARY ITS: ONLY BUS CONTROLLERS HAVE ASYNCHRONOUS MESSAGES NO. MESSAGES/PROCESSOR CYCLE=10

Figure 3-2. Percent of Bus Utilization for Different ITS Configurations

The BIU complexity was examined for each ITS. The stationary master and nonstationary master BIU designs consist of two LSI chips. The contention BIU is the most complex and would take a minimum of four LSI chips to implement.

The primary advantages and disadvantages of each ITS are compared in figure 3-3.

3.1.2 DAIS Change Recommendations to Support a Hierarchical, Distributed Processing System Architecture

A major goal of the AASMMA program was to study the use of a system architecture that achieves independent operations between the various levels of physical and logical structures. The original DAIS concept was limited to the study of DAIS single-level minicomputer systems but this study was to expand the DAIS concept into multi-level systems. Three aspects of the AASMMA program are applied to DAIS baseline to determine which changes should be made. The first aspect is the distributed processing organization made possible by the availability of relatively inexpensive microprocessors that will replace conventional hardwired sensors and remote terminals as well as minicomputers used to process the data. The second aspect is the application of a hierarchical organization of multiple buses organized to create independent subsystems which are replaceable on a mission basis. The last aspect is to determine potential changes to DAIS baseline if it were to use one of the ITS developed in the AASMMA study to achieve multimission independence.

Changes to the existing DAIS standards to permit distributed processing include: 1) increasing the capability of the master executive to handle more than four ("n") intelligent processing elements (limiting the executive to handling four processors is not a conceptual limit, rather it is a limitation of the initial DAIS executive implementation); 2) modifying the executive support software (PALEFAC) to enable partitioning of software to the "n" processing elements; 3) modifying the local executive to operate in remote terminals and to perform sensor input and output; 4) increase the local executive's modularity to allow only the necessary functions to be used in intelligent sensors rather than installing a larger more powerful general purpose executive; 5) increase the master executive's modularity to allow for both single processor and synchronous executive control.

Because the original DAIS program was implemented using a single BIU for the communications medium, hardware and software changes are required to accomodate hierarchical networks. These changes include: 1) allowing two or more BIUs to communicate with a single processing element (PE); 2) updating the local executive to respond to multiple BIU updates (minor cycle event) independently; and 3) modifying executive support software to structure compools such that each BIU can access its own set of data and commands.

	Stationary master	Nonstationary master	Contention
Description:	DAIS control procedures were modified to include MIL-STD-1553B and multiple bus interfaces, but is essentially DAIS in concept.	DAIS control procedures were modified to include MIL-STD-1553B, bus control allocation and multiple bus interface control.	New control procedures are required as well as a new bus protocol. The operation of the system is significantly different from the other two ITS.
BIU:	Simple/reliable BIU. One is already in production which would suffice (2 chips)	Polling message sequence causes high processor overhead or complex BIU: Same as the stationary master (2 chips)	Complex; (the 4 chip BIU would be much more complex than the 2 chip BIU)
Risk of development:	Minimal (DAIS is the base from which changes would be made)	Low (can be built on DAIS concepts with more changes than stationary master)	Hardware/software development; asynchronously operating software is more difficult to integrate than synchronously operating software; DAIS local executive can be used
Control dependability:	Control is localized to one location	Greater independence than stationary master, with less control dependability due to transfer of master	Greatest independence of functions
Bus efficiency:	Overhead is primarily only concerned with message movement	Worse than stationary master and contention	Low efficiency with high collisions only if the system is not tuned properly
Synchronous message processing overhead:	Efficient synchronous message processing	Same as stationary master	Not applicable - all message processing asynchronous
Time critical responses:	Asynchronous(trigger) messages require longer wait and processing times than the other two ITS	Trigger messages can be output faster than stationary master	Much faster than the other two ITS
Multi mission flexibility:	For any given change of mission, the master must be altered to accommodate a new message set	Each mission controlled through a different master allows core processing to remain unchanged	Changes occur by replacing equipment with other equipment
Functional enhancement:	Any function added to the distributed system would require a change to the executive tables	The addition of functions would normally perturb the message structure unless the new messages occupied previously unused bus time.	Functions added only affect the processor in which the function is added

Figure 3-3. Candidate Information Transfer System Comparison

The changes in the present DAIS implementation to support various information transfer systems involve changing the master executive program. Since the master executive contains the specific functions of transmission of messages, error handling, synchronization and asynchronous message control, it will be highly dependent on the information transfer system selected. The local executive is independent of the type of ITS and consequently it will remain essentially unchanged.

In conclusion, an analysis of existing software standards indicates that conceptually the DAIS approach is viable and can contribute to the overall effectiveness of software development activities for a multimission avionics system. The design methodology, architectural standards and coding methodology defined by DAIS can be used unchanged in microprocessor based systems. The concept of a modular standard executive computer program and enforced software interfaces initiated by the DAIS program is an effective method for simplifying the overall development activity, particularly in the area of integration. The DAIS standards include the requirement that the application software be independent of all interprocessor communication. The purpose of this standard is to isolate the application software from the ITS, thus allowing all application software to be used without modification regardless of the ITS. The executive (an integral part of each ITS) will have some modules that are common to DAIS and the ITS selected for multimission hierarchical structures, and some modules that are unique to each ITS.

3.1.3 Microprocessor Technology Forecast

A technology forecast was conducted in the AASMMA study in 1978, and the forecast covered the time to the year 1982. The study revealed that the semiconductor industry is producing high performance components which are inexpensive and highly reliable. These high density parts are moving from the realm of LSI (large scale integration) to VLSI (very large scale integration) with an effective doubling in chip capacity observed each year. Monolithic microprocessors are now at the head of the semiconductor boom. The monolith-based microcomputer offers the advantage of lower cost, weight, volume and power as a result of fewer components. The lower cost encourages the use of a distributed system since lower per unit costs are normally reflected in a lower cost of integration. A summary of the technology forecast is shown in figure 3-4.

The 16-bit microprocessor will be prevalent in the early 1980's. It will provide adequate computer power (in the range of today's high-end minis), precision and software maturity for avionic applications. Advanced 32-bit microprocessors are expected to be available; however, their immediate use in 1982 avionic systems will be limited to specialized applications. One key reason for the 32-bit forecast is the anticipated lack of application software, since most avionics software is directed toward the 16-bit machine and no events are anticipated by 1982 to change this trend. Advanced microprocessor architectures will appear using multiple processing units on a chip to handle communication or arithmetic processing in parallel with the ALU. Due to the requirements for military-qualified components, it is doubtful that many such components will find their way into avionic systems by 1982. The bit-slice elements will be used:

	Cost	Architecture impact	Standards
Microprocessor	Monolith microprocessor will be much more cost effective than bit slice. Nuclear hardness requirements preclude NMOS monoliths but allow bi-polar bit slice. 16-bit monoliths will cost \$20 - \$30 commercially.	Micros of 1982 will be equivalent to minis of today. JAN qualified micro will be limited to one or two 16-bit machines.	Monolith MIL-STD-1750 will not be available unless government initiates design and development. BIU to interface MIL-STD-1553 will remain at the cost of limited production chips.
Memory	Capacity will grow for the same size and cost of memory.	Magnetic bubble technology will not be available in 1982 because of temperature limitations.	1982 MIL -qualified memory standards include: 256 K ROM 64 K EPROM 16 K STATIC RAM 64 K DYNAMIC RAM 256 K CCD

Figure 3-4. 1982 Technology Forecast

1) for special purpose applications,
2) to emulate a computer whose architecture is not available in the monoliths (e.g. MIL-STD-1750), and
3) to satisfy some nuclear hardness requirements using bi-polar components.

Shielding against high energy particles is fairly ineffective since it requires several inches of lead for some environments. The shielding approach is prohibited for this reason for some avionic applications. In general, shielding is not practical and the fabricating technology will itself have to satisfy stringent nuclear requirements. Some I^2L monolithic microprocessors are available, but they are not as hard as TTL components.

Military qualification of microprocessor chips is extremely expensive. The mechanical qualification of the chips is well defined and qualification of production techniques is somewhat standard. The difficulty is in the development of the electrical functional tests which are used to qualify the electronic operation. Each test must be tailored to the individual machine. To date, a test has been developed for the INTEL 8080 and one is under development for the RCA 1802. Barring any major DoD funding changes that would shorten the MIL-qualification process, it is anticipated that one, perhaps two, 16-bit microcomputers will be military-qualified by 1982. The availability of military-qualified components is necessary to obtain flight qualified LRUs. However, the issue of selecting a military-qualified monolith versus a military-qualified bit-slice does not appear to impact the qualification requirements of the sensor with an embedded microprocessor. The functional test requirements appear to be geared to the box specification requirements and not to the way the internal electronics are assembled and tested.

In general, new 16-bit military-qualified monolithic microprocessors that are as powerful as existing minicomputers are expected to be available for use in avionic applications. It is doubtful that the avionic microprocessors of 1982 will conform to any special mandated standards, but rather a defacto standard will evolve from the selected microprocessor. For mandated standards either a bit-slice approach, which is much more complex, or a specially built monolith will be required. The nuclear hardness issue prohibits NMOS for all but extremely benign applications. CMOS/SOS, I^2L and TTL are better able to handle nuclear hardness requirements.

In the memory area, bubble memory temperature problems will probably prevent military-qualification before 1982; however, progress is rapid and the interest is sufficient to believe that bubbles will soon be available for military use. The volatile storage CCD is akin to NMOS and their nuclear hardness capability is nil. EPROM will be available in 64K bit NMOS military-qualified packages and 16K bit CMOS military-qualified packages. For nuclear environments 16K bipolar military-qualified PROM will be available. In addition, CMOS/SOS will be available in 16K static and 4K dynamic RAM.

In the BIU area, special chips are being manufactured to satisfy MIL-STD-1553 requirements. It is anticipated that these components will be military-qualified by 1982. Also, BIU modules using bit-slice technology are available. In either case, the limited commercial usage of MIL-STD-1553 precludes the very low prices associated with volume production.

3.2 TASK II. COST MODELING SUMMARY

The primary evaluation criterion used on any military program is cost when performance has been defined as acceptable. Four parametric cost models were acquired or developed to provide the means of systematically evaluating the costs of proposed hardware and software configurations. These models are discussed extensively in appendices G and H of the First Interim Report. The RCA system of PRICE models was selected as the most appropriate set of tools to estimate the costs of hardware acquisition, hardware maintenance and software acquisition. The software maintenance cost model was developed by Boeing specifically for this study because no existing models had the necessary sensitivity for this application. These tools were to be used to evaluate the information transfer systems and standards and not to be applied in any other application. The effectiveness of these cost models were not demonstrated because the evaluation phase of this contract was deleted.

The PRICE hardware acquisition model has been used for many years by both military and commercial organizations and has become the standard hardware cost model. The PRICE hardware LCC model is a natural extension of the PRICE hardware acquisition model.

The selection of software cost models was a more difficult task. The basic prediction used in establishing software cost is the number of instructions in the program. Models that used only this parameter were found to be inconsistent. Orders of magnitude errors were experienced in some cases. Other software characteristics (e.g. complexity, type of language) were integrated in a variety of cost models. These models often made better predictions than the single parameter models, but their accuracies were still questionable. PRICE S was found to be the most acceptable parametric model for estimating software development cost using the salient parameters of the AASMMA study. In the cases tested it made predictions that were accurate to within 20% of actuals in all but one case, in which it was in error by a factor of 2. Approximately twenty five runs were made using PRICE S.

Very little information is available on the life cycle costs of software. The existing software LCC models are very simplistic and did not contain sufficient sensitivity to be useful in the AASMMA study. For example, one model predicts a requirement of one man for each 10,000 instructions while another predicts the software support requirements to be a productivity factor applied to the expected number of instructions that are changed each year. The software LCC model had to be sensitive to the study parameters of language type, transferability, support software requirements and distribution of the processing load. Since such a model did not exist, one was developed.* The accuracy of this model cannot be validated because of the paucity of software LCC data. However, predictions made by this model are consistent with predictions made by the other software LCC models. The most sensitive parameters which impact the life cycle cost are: (1) change rate, (2) operational years, (3) size, and (4) transferability. Note that the transferability issue is the only criterion addressed in this study.

* Troth, Frank and McSharry, Mike. "Users Manual and Technical Description of BMAD Life Cycle Cost Model" 24 July 1979. Boeing Document #D180-25095-1.

3.3 TASK III. STANDARDS INVESTIGATION SUMMARY

3.3.1 Hardware Standards

The standardization of hardware can be performed at many levels ranging from the standard component level to the standard system level. However, standards at the low levels tend to restrict technology and innovative design. The lowest level standard that is reasonably independent of technology is that of the functional module. Use of this functional module standard, a network bus standard, and a subsystem interface standard would provide an effective group of hardware standards. The network bus standard is expected to continue along MIL-STD-1553 trends due to the firm precedent already established by this standard and the lack of a commercially acceptable alternative for avionic applications. No subsystem interface (i.e. back-plane or back-bus) standard is in existence for military applications. Selection of a subsystem interface from the set of commercial back-bus standards (e.g., IEEE 488, Intel Multibus, IEEE S-100) appears to be the most cost effective approach. The standards associated with functional modules would cover microprocessor, memory, sensor I/O and special processing devices. Candidates for the microprocessor standards include the MIL-STD-1750 instruction set and popular defacto commercial standards (e.g. Z8000, LSI-11, Intel 8086). Memory modules are to a large extent functionally specified by a back-bus standard. The sensor I/O interface module will have requirements to interface with parallel digital, serial digital, analog syncro and pulse trains. While it is felt that a set of definitions such as for the DAIS interface modules is possible for these interfaces, it is not necessary because sensor inputs/outputs will generally be built to interface directly with the processor. Consequently, the definition of a back-bus will provide the necessary definition of all sensor inputs/outputs.

3.3.2 Software Standards

Like hardware standards, software standards can be applied at the module definition and interface level. Language software standards control the interface between the software and the designer, integrator and maintenance personnel. An HOL standard has been proven to be an effective way to reduce costs, and an HOL is as applicable to a microcomputer system as a minicomputer system. Application of a standard instruction set will reduce support software costs, but it is not clear whether or not these software related costs will offset the potentially increased cost of hardware. Module software design standards, such as top-down structure and modern programming practices, control the module configuration and the interfaces to the module. The most significant finding in the software standards study is that the DAIS concept of standardizing the executive program and the application software structure is potentially more cost effective than any other proposed standard (HOL, standard instruction set, etc.) and is directly applicable to microcomputer based distributed architectures. However, the existing DAIS executive itself must be modified to operate in a distributed, hierarchical processing network. The recommended changes are summarized in paragraph 4.2, and described in detail in the first Interim Report (Appendix E), and the stationary master part one specification.

4.0 DETAILED PROGRAM FINDINGS

4.1 CANDIDATE INFORMATION TRANSFER SYSTEMS FOR MULTIMISSION APPLICATIONS

In Task I, three candidate ITS were designed for future applications. They are: the stationary master, the nonstationary master and the contention multiple access information transfer systems. These three ITS are described and compared below. For more detailed descriptions and comparisons of the Task I derived ITS see Appendices A, B, C and D of the first Interim Report, Volume II. System Control Procedures and Part One Specifications for each of these ITS were written as part of Task IV and are summarized in Volume II of this report.

4.1.1 Description of the Information Transfer Systems

The stationary master information transfer system (SMITS) is similar to DAIS in concept. The control site is centralized in one preselected terminal, designated the master bus controller, and this control point may be relocated to another terminal, designated the monitor bus controller, in the event of a master bus controller failure. Some of the advantages of the stationary master ITS are a simple/reliable BIU, minimal risk of development, control dependability, effective bus capacity, low processor overhead for synchronous messages, use of the MIL-STD-1553B protocol and similarity to the DAIS ITS. The multi-protocol issue (several different versions of 1553) has been solved by using a BIU which can interpret status bits depending upon the address of the message, and interrupt the PE only when required. Such a BIU implementation would allow previously developed equipment to be used with current 1553B equipment. The DAIS system specifications and standards previously developed were used as a baseline and were modified to provide the stationary master concept identified here. Some of the disadvantages of the stationary master ITS are multimission inflexibility, time critical responses and limited functional enhancement. The stationary master in general represents a low risk to the overall development of an operating system for an aircraft.

The nonstationary master information transfer system is based upon modified DAIS system control procedures developed for the MIL-STD-1553A bus. These procedures have been modified to support a nonstationary master architecture which uses a polling scheme to determine the processor that will next control the bus. In addition, the protocol features of MIL-STD-1553B have been incorporated. The result is a system which closely resembles DAIS in its philosophy and application, but provides additional capabilities for the multimission applications envisioned for advanced digital systems in future aircraft.

As derived in Task I, the nonstationary master system utilizes polling to provide the system designer increased flexibility in determining which mission capability is resident in a given processor. For example, a mission change (e.g., a reconfiguration from a reconnaissance mission to an attack mission) can be accomplished in the nonstationary master system without modifying any of the bus control tables or procedures. The only overt action needed to implement the change is to load into the aircraft the multimission processor with software appropriate for the new mission and interconnect the new avionic subsystems. This change is completely transparent to the rest of the system. Therefore, all mission dependent bus messages are initiated by the multimission processor when it gains control of the bus. The multimission processor uses standard bus commands to collect the necessary data from the other devices on the bus. Part of the nonstationary philosophy includes a core of processors which perform the general processing functions and multimission processors which will be changed as required for new missions. The advantage of this philosophy is that the core of processors is isolated from the multimission processors, which minimizes change from mission to mission.

The polling scheme developed for nonstationary master ITS differs from most bus polling in that there is no fixed bus allocation sequence. After each master has completed its required bus activity or has used up its preset maximum amount of time as master, it must poll the other potential bus controllers and decide which processing element is to be the next master. The information sent to the current master indicates the need for bus use by each processor. This method provides a means for high priority messages, regardless of processor location, to be transmitted before most lower priority messages. This scheme minimizes the system reaction time and reduces the long delay times usually associated with most round-robin or structured polling schemes. (It should be noted here that the bus control transfer protocol for the nonstationary master ITS documented in Task IV is a round-robin scheme. The reason for the change is the realization that in such a multimission hierarchy, only one or two multimission controllers would be used, and polling would be unnecessary.)

The nonstationary master scheme has been developed out of the technology available from the stationary master DAIS system. The DAIS system would require a new mechanism to speed up bus control transfer between controlling processors. With this change and limited changes to the specifications, standards and software already developed for DAIS, a nonstationary ITS could be developed which would significantly improve the ability to support a multimission application.

In the contention multiple access information transfer system, there are no unique bus masters, rather each device contends for use of the transmission media. When a device wishes to transmit, it will sense the communication activity on the transmission media to determine if a transmission is allowed. If activity is detected, the device will wait until the transmission ceases, then the device will begin a random delay associated with its queued message's priority before attempting to transmit. If the transmission media has remained free of communication activity for the duration of the delay, the device initiates transmission. Utilizing this approach it is possible for multiple devices to attempt communications on the transmission media simultaneously. If this occurs, a collision is said to have occurred, and the transmissions are terminated and rescheduled for transmission later. Once a device acquires the transmission media and begins communication which does not result in a collision, it may transmit until a given length of time expires. During this period of time, called the transmission interval, as many complete contiguous messages as can be transmitted in the time interval will be scheduled and transmitted.

The contention mechanism discussed here allows considerable flexibility in system integration. This ITS has the advantage of easily providing for new, modified or multimission oriented sensors. Because of the independent nature of the processors, modifications to the system will primarily affect the processors responsible for control of the modified section.

Each of the messages are addressed by content rather than by origin or destination and multiple BIU's can accept the same messages. Each message can be considered to be broadcasted in a source oriented system. Once a message arrives at a destination, the arrival can create an event which may cause the invocation of a task waiting for the data. The operation of such a transmission system is primarily asynchronous in nature. Any cyclic operations are scheduled to be performed by the executive using the processing element's local clock. This local synchronism, coupled with time tagging of sensor data, can potentially provide more accurate data to functions requiring data generated on a time oriented basis than systems that operate on a minor cycle basis. However, using this method of timing does not preclude the use of minor cycles, which could be used to synchronize local processing element clocks and control the transmission of data on a periodic minor cycle basis (Analysis has shown this contention system is capable of completing a required set of data transmissions more quickly in a minor cycle than the stationary master). If a specific message in response is required, a request to deliver this data is provided. A basic assumption in the use of this mechanism is that few messages require requests. The system has the ability to interrupt contention and act in a manner similar to a master/slave ITS when required (i.e., an error handling message is required).

There are several advantages provided by the contention system. The first of these is that the system is the easiest system to add multimission functions. Each new function could potentially accept the core avionics data without impact to the system, since data is transmitted in a source oriented fashion. The data associated with mission specific functions would be generated by contending for the bus and transmitting the appropriate data to its own subsystems and to integration and display functions. The upgrading of an existing contention system, by adding new functions, can be done easily because of the asynchronous operation of the ITS. Functions by necessity and design are as loosely coupled as possible. This loose coupling will allow for easier upgrade of capability than systems that are tightly coupled by the information transfer system using synchronously generated messages within a given period of time. This contention organization also is most closely aligned with the method of integration used today by an avionics integrator who purchases subsystems and integrates them using the ITS integration process. Because the contention algorithms are based on message priority, the highest priority messages can be transmitted within a shorter time than other type multiple control mechanisms. This capability will be very important in advanced vehicles where rapid responses are required.

Several drawbacks to the contention system have been identified. These include the somewhat asynchronous nature of the system (even using minor cycles). Difficulties occur in the debugging of the system during initial integration because error conditions are not easily repeatable. This asynchronous system also could potentially require different control algorithms than are normally used in totally synchronous systems. These algorithms do not allow the simplifying assumptions usually made in time dependent algorithms. The advantage of time tagged data is that it does provide the capability to modify data which is collected synchronously and transmitted asynchronously. The contention system bus allocation algorithms involve the detection of a inactive bus, a random waiting period and the transmitting of a message sequence or the bus. This procedure requires less bus overhead than a stationary master system.

4.1.2 Analysis of Information Transfer Systems

The analysis performed in support of this Phase I task was updated in support of Task IV of Phase II. The updated material is presented in paragraph 4.2 of Volume 2 of the final report. The previous analyses were given in appendices A, B, C and D of the First Interim Report.

4.2 RECOMMENDED DAIS CHANGES

4.2.1 Alterations to DAIS Standards for Distributed Processing

The introduction of mircoprocessors to the DAIS architecture can be accomplished with very minor changes to the master and local executives, and with no change to application software. The PALEFAC support tool will have to be updated to reflect the distributed processing network.

The current DAIS master executive is capable of handling up to four intelligent processing elements. This is not a conceptual limit, but is rather a design limitation of the initial DAIS executive. To bring DAIS up to an "n" processor system which is appropriate for distributed processing requires only a minor internal modification. This same modification must also be reflected in the PALEFAC support utility to enable it to partition software into "n" processing elements.

The local executive also requires modification to be able to operate in intelligent remote terminals. In a highly distributed configuration the processing elements will perform only a small number of functions. they will interface with a sensor(s), process the sensor's information and transmit it across the bus. The current DAIS processing elements have no mechanism to communicate with sensors. All sensor communication is done via remote terminals.

The current DAIS local executive may have to be changed to handle interrupts from both the BIU and a sensor. Even if the sensors do not utilize interrupts to signal data transfer, information must be transferred to/from the microprocessors. This must be accomplished by either the local executive or application programs. The DAIS standard of environment independence for application programs must be compromised if application programs are assigned I/O responsibility. This alternative will not be considered for the distributed systems. This leaves the responsibility with the local executive. The local executive must be changed to accept I/O from devices other than a BIU.

Another change to DAIS which would aid a distributed processing hierarchical organization is additional modularity. All processing elements currently contain identical copies of the local executive. As the processing elements become more specialized, the local executive supporting the processing element could also become more specialized. Installing a powerful general purpose executive in each microprocessor could introduce unnecessary overhead and makes the use of the DAIS executive undesirable to subsystem manufacturers. The DAIS local executive should be further modularized so that only the necessary modules be included in any given processing element. Table 4-1 contains a possible modularization and extension of the current DAIS executive programs to support more distributed and specialized systems. The comparison with DAIS is shown in table 4-2.

This modular application should result in a less complex executive program in each processing element because only necessary software functional elements would be included. Additionally, no assumptions should be made relative to I/O devices in the basic core executive.

4.2.2 Alterations to DAIS for Hierarchical Networks

Changes to DAIS due to the addition of a physical hierarchical structure (figure 4-1) would have to be made in both hardware and software. The hardware changes are necessary since two or more BIUs would communicate with the same processing element. The BIU interfaces require that multiple DMA's be available to input or output data at a rate sufficient to meet the ITS transfer requirement. The DAIS processor was built with the assumption that a single BIU would be the communication medium. The interrupt structure and the software must be modified to allow multiple BIUs to operate in any combination of master or local processing elements. Currently, a specific interrupt implies that a specific master executive or local executive routine is involved in handling the interrupt. If a processing element is a controller for two levels or if a processing element is a remote for two levels the interrupts are currently not arranged properly to service these identical functions.

The software changes which must be made due to a physical hierarchical organization is primarily in the modularity and reentrance capability of specific functions. The two specific items which need to change are the local and master executives. The local executive needs minor changes. If the local executive is interfaced with two buses, then it must respond to minor cycle updates from both levels. Normally, a minor cycle update implies that message addresses are changed for the BIU and tasks are scheduled according to the minor cycle event. These functions must now be done independently for each bus. Compools must also be arranged so that each BIU can access its own set of data. This problem can be handled by appropriate modifications to the support software (PALEFAC). If two master executives are resident in the same processing element, then a single clock should be controlling both executives. Each executive must be sufficiently modular to handle independent sets of asynchronous message requests and should communicate minor cycle events to the single local executive independently.

Table 4-1 POTENTIAL MODULARIZATION

FUNCTIONAL ELEMENT	FUNCTION
A. BUS CONTROL PROGRAM	Controls all bus activity; fields request for bus activity, error handling and time synchronization.
B. INTER-PROCESSOR REQUEST PROGRAM	Assemblies inter-processor requests into valid bus messages and communicates with Bus Control Program.
C. ASYNCHRONOUS RECEIVE PROGRAM	Handles the reception of asynchronous messages, queue manipulation and message dispersal.
D. ASYNCHRONOUS TRANSMIT PROGRAM	Handles the transmission of asynchronous messages and queue manipulation in response to requests from Task Service Program.
E. TASK CONTROL PROGRAM	Handles scheduling termination and task execution sequence fo:ʳ applications programs in this PE.
F. TASK SERVICE PROGRAM	Handles service requests from tasks in this PE, such as COMPOOL or DATA access, timing, wait requests and event signals.
G. BUS LISTEN/RESPOND PROGRAM	Handles any necessary response to bus messages such as DMA pointer set up as a result of a Master Function Request (i.e., time synch.) This is a remote BIU interface program.
H. LOCAL I/O PROGRAM	Handles nonbus I/O such as reading sensor inputs, writing of corrections to sensors or writing to an integrated C&D.

Table 4-2. DAIS COMPARISON

Dais functional element	Proposed functional element
Bus control program	Bus control program
Local executive program	Bus listen/respond program
	Interprocessor request program
	Asynchronous receive program
	Asynchronous transmit program
	Task control program
	Task service program
Not available	Local I/O program

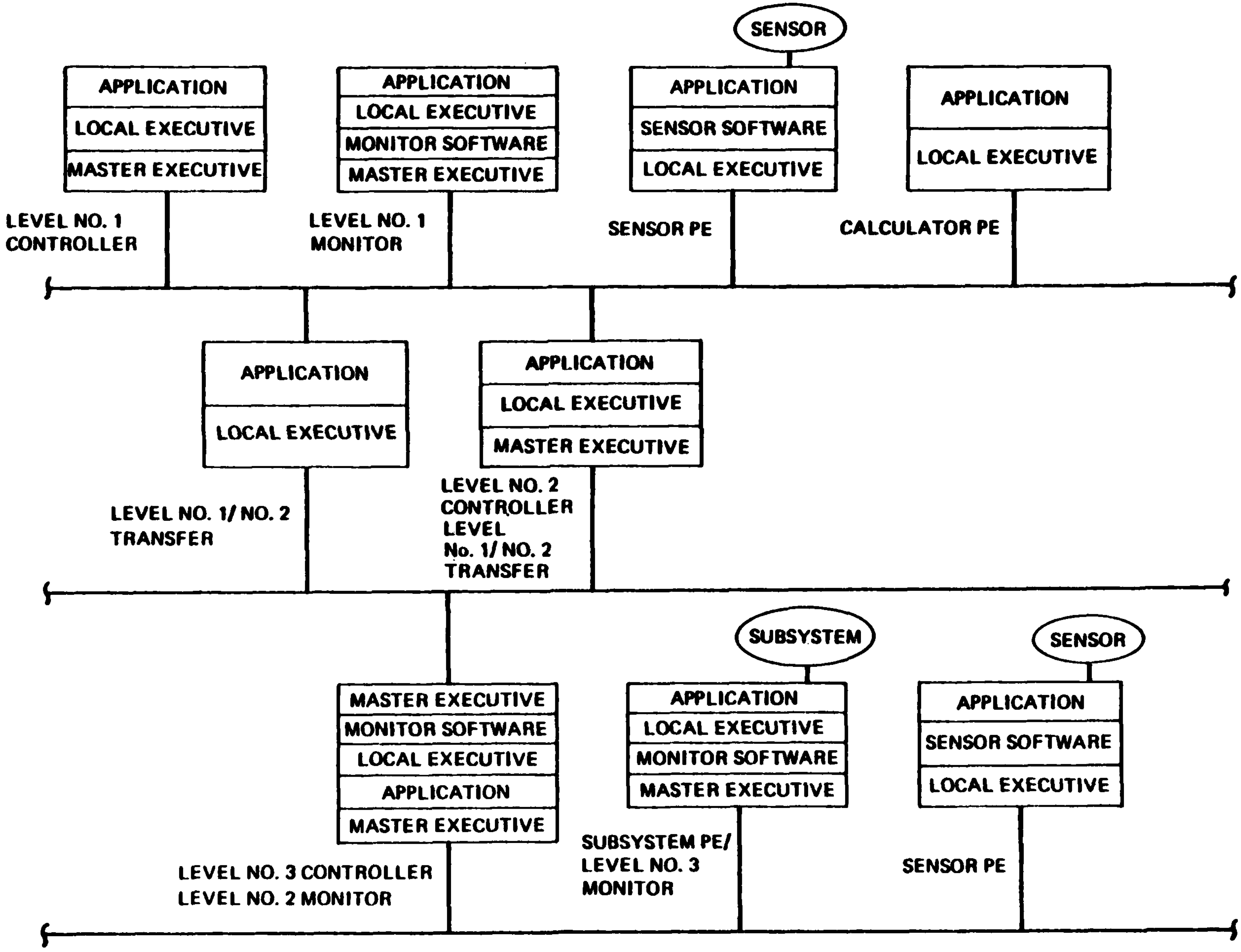


Figure 4-1. Hierarchical Structure

4.2.3 DAIS Changes to Support Each AASMMA Information Transfer System

An analysis of existing software standards indicates that conceptually, the DAIS approach is quite viable and can contribute to the overall effectiveness of software development activities for a multimission avionic system. The design methodology, architectural standards and coding methodology could be used. Since these methodologies address discipline in design, structure and development, they could be successfully applied to software development of distributed microprocessor based systems.

The concept of a modular executive computer program and enforced interfaces initiated by the DAIS program for avionics, is an effective method for overall development activity particularly in the area of integration. As such, this concept should be employed for development of the application and executive software. The DAIS standards require the application software to be independent of all interprocessor communication. This standard isolates the application software from the ITS, thus allowing application software to be used without modification on all three ITS. The executive (being an integral part of each ITS) will have some modules that are common among DAIS and the three ITS described in the AASMMA program, and some modules that are unique to each ITS.

The master executive functions are unique to the ITS and require changes as the design of the information transfer system changes. The master executive contains the ITS specific functions of: transmission of messages, ITS error handling (such as transmission and equipment status changes), ITS minor cycle synchronization and asynchronous message control.

The DAIS executive, in its current state, supports a stationary single point control with multiple processors and the DAIS master executive can be applied directly to the stationary master information transfer system. The DAIS local executive can be used with the exception that (1) any hardware-specific interfaces to the BIU (and master executive) might be required to change, and (2) a sensor I/O interface may be required to replace the remote terminal concept with processing elements and a simple local executive. While the local executive could be used in its current state, the addition of simpler modules could be made to accommodate the required functions. A candidate list of modules was previously listed in table 4-1.

The nonstationary master executive differs from the DAIS master executive in the following areas:

1) Transfer and acceptance of bus control as a normal operation
2) A single nonstationary master is responsible for minor cycle synchronization
3) Responsible for monitoring and detecting a bus control transfer failure

The first change is predicated on the use of the stationary master BIU. A nonstationary master BIU could and should be designed to perform this function in hardware, leaving the nonstationary master executive almost identical to the stationary master executive.

The contention master executive requires another separate set of changes to the DAIS master executive. The master executive can take on two forms depending on the type of organization that the system designer desires: synchronous and asynchronous.

The synchronous form of contention provides a minor cycle update function in one given controller which keeps system time similar to the DAIS master executive. The DAIS master executive knows the transmission status of the system based on whether or not the master BIU has interrupted to indicate the end of the message transmission list. The synchronous contention master executive would have to receive completion messages from all active elements to know whether a minor cycle had been completed. Consequently, this completion function must be a part of the master executive. Each processing element would contain a master executive so that each processing element would be responsible for the transmission of its own asynchronous messages. The awkward asynchronous message transmission mechanism used by DAIS would not be required.

The completely asynchronous contention master executive requires a clock in each processing element. The clock can be used to run a synchronous local executive and can be used to time-tag data to give an accurate accounting of the time when the data was collected. In this case, the arrival of data would be strictly asynchronous. Efficient mechanisms in the master executive would be required to turn the message arrivals into events which the local executive could use. Buffer management would also be required by the master executive, because multiple asynchronous messages of identical identification could arrive before being accepted by the local executive. In addition, messages may be asynchronously generated or requested as in the case of trigger messages. In the asynchronous contention system, the only difference between normal asynchronous messages and trigger messages are the priorities.

4.3 TECHNOLOGY FORECAST

The purpose of this forecast is to predict those components (microprocessors, memories and BIU's) that are likely to be available for use in the design of an avionic systems in the early 1980's. The forecast addresses not only which components will be available, but also their cost and performance. The forecast is specifically for the 1982 time frame and it deals with those devices that are projected to be available for use by a system designer. It is assumed that military qualified devices will grow out of commercial lines, since this has been the trend in the past. A device is typically first developed for the commercial market, and later "militarized", or Mil-spec qualified (e.g., Intel 8080 microprocessor). The list of eighteen references used to compile the technology forecast is found in appendix F of the First Interim Report.

While the military will fund development of a few specialized chips, it is felt that this will be the extent of their impact. In general, the Mil-qualified process does not appear to be a problem. Nuclear hardness issues prevent the use of NMOS components for all but the most benign applications. The I^2L, TTL or CMOS/SOS technologies better satisfy nuclear hardness requirements.

The conclusion of the technology forecast is that VLSI technology is here. Powerful 16-bit microcomputers, even faster than presently used minicomputers, have arrived. Memory technology has increased bits per chip to highly dense packages and will continue. By 1982 the component technology for distributed microcomputer based systems will have matured.

4.3.1 Microprocessor Technology

Two groups of microprocessors are analyzed for the 1982 time frame: monolithic microprocessors and bit-slice microprocessors. Obviously, the monolithic microprocessor has a lower cost than the bit-slice microprocessor but the bit-slice machine will continue to provide higher speed, emulation capability and special processing capability. It is anticipated that the candidate monolithic microprocessor will be a 16-bit Mil-qualified component whose operation will exceed the capability of present day minicomputers such as the PDP 11/45. The major unknowns are the effects of standards and nuclear hardness. It is felt that heavy military involvement is necessary to evolve a processor. Without this involvement, a defacto standard (i.e., 8086, etc.) will evolve. Monolithic multiprocessor technology will be highly developed; however, it is felt that the lack of Mil-qualified components will impair their availability for avionic applications. If the commercial market fails to deliver hardened components, then the military must again become involved. It is anticipated by 1982 that special processing requirements and their associated chips will evolve to complement the microprocessor as a potential candidate for all avionic processing.

4.3.1.1 Monolithic Microprocessors

Popular monoliths are found in 8 and 16-bit configurations, with the latter being the relative newcomer to the market. The 8-bit machine, now highly matured as both a control and arithmetic unit, will continue as a cost effective solution to medium complexity applications. This device is solidly entrenched in several arithmetic processing systems; however, a gradual slip due to the 16-bit entries is seen in this market. The 16-bit machine has many clear advantages over 8-bit architectures and will become the industry standard by 1982.

The future acceptance of the 16-bit micros may be extrapolated from the current success of the 16-bit mini. Sixteen-bit computers are meeting todays needs by providing adequate precision and computer power for most applications. Existing minicomputer software may be used in many cases on 16-bit microprocessors with compatible architectures. Although the appearance of 32-bit microprocessors is expected by 1982, these machines will be limited at first to a selective market requiring high performance and precision. Any major trends to 32-bit architectures is expected only after 32-bit minicomputers have become more popular. Recent announcements of new 16-bit devices show a growing maturity in architectures which is expected to continue. The Intel 8086, Zilog Z-8000 and Motorola 68000 collectively demonstrate most of the features anticipated for a standard 1982 machine. Enhancements between now and 1982 are expected to consolidate these capabilities in a single processor as well as exploring the multiprocessor architectures. In general, the 1982 monolith will greatly resemble todays minicomputer. Double precision integer arithmetic (including multiply and divide) will be hardware instructions. Direct addressing of several million bytes of memory will be provided, which should be more than sufficient for most applications. Address modes will include those common to many minicomputers architectures, including direct, indexed, indirect, autoincrement and autodecrement. Selectable interrupt modes will be provided including both maskable and nonmaskable types along with vectored interrupt capabilities. Larger amount of memory will appear on the chip as the level of chip integration increases. The mix of ROM and RAM will vary, with a clear split between ROM and RAM dominant chips appearing. The ROM-dominant chips with up to 32K bytes ROM and 2K bytes RAM by 1982, will appeal to those applications with fixed programs and desiring minimum part systems. Applications desiring more RAM to use as a fast scratch pad area will utilize the RAM-dominant parts, with up to 8K RAM and nominal ROM available on the chip. The technique of memory mapped I/O will virtually end I/O space restrictions. Features such as the Read-Modify-Write command and similar semaphore manipulations will facilitate the use of shared I/O devices in multiprocessor systems.

The most significant development in the 16-bit microprocessor of 1982 will probably be the migration of special processes (in the form of "smart" controllers) onto a single microprocessor chip. These functions will operate as separate processing logic operating simultaneously and asynchronously to the other on-chip processing elements. The particular processor groupings will be determined by chip real estate and packaging constraints. Naturally, all combinations of functional partitioning will not be available. Instead, broad groupings will form creating several "classes" of monoliths, such as: I/O and arithmetic processors for data acquisition; floating point and memory management processors for scientific and engineering applications; and communications processors, protocol handlers and bus management for message switching, control or network applications.

These multiprocessor architectures will naturally be classed in the high performance range of miniature (mini and micro) computers. While the task of integrating these functions onto a single VLSI chip and managing pinout and packaging problems are indeed challenging, the organization of the software is perhaps a more serious problem. Since a single distributed system would likely require a mix of these monolith classes, it would be desirable to have software compatibility between the classifications. It is anticipated that a common instruction repertoire could be adopted composed of a superset of commands required for each class of monolith. In this scheme, instructions not relating to processors configured on a particular chip could be trapped and a branch made to emulate the operation in software.

The monolithic microprocessor will continue to satisfy the large majority of applications. As the level of integration increases, other technologies are expected to be used for monoliths. Bipolar monoliths will invade the high end of the market particularly as specialized machines. The CMOS/SOS technology is expected to have an even greater impact due to its low power consumption and near-bipolar speeds in addition to its nuclear hardness capability.

4.3.1.2 Bit-Slice Microprocessors

In contrast to the fixed architectures of monolith machines, the bit-slice elements rely on the designer to define and implement the processor architecture. The macro instruction set is completely definable through microprogramming, making the bit-slice microprocessor ideal for emulation. Furthermore, bit-slice elements are usually on the order of ten times faster than their monolith counterparts, which makes them well suited for special high performance applications. Further speed gains may be obtained through parallel or pipelined constructions using the bit slice primitives.

Bit slice components are usually composed of bipolar materials, with Schottky TTL the most popular. The Motorola 10800 uses ultra-fast ECL technology; however, its application is limited due primarily to its incompatibility with other technology interfaces. The newly announced 8-bit slice CPU uses CMOS/SOS material, which makes it attractive due to its low power and environmental immunities. This technology is expected to rival bipolar processes as speed improvements are made. The use of bipolar technology is one reason why bit slice devices have not yet experienced as great an impact from VLSI advances, in that the VLSI emphasis has been toward MOS technology.

The 1982 bit slice families will resemble to a great extent the bit slice elements of today. The 4-bit slice component is still expected to be the industry standard in 1982. A primary reason for this is attributed to the difficulty of using bit-slice components, requiring a delicate mixture of both hardware and software design skills. This causes a slower design turnover for newer devices. Larger bit-slices like the 8-bit devices are appearing now and a new 16-bit slice expected by 1982. The thrust of the efforts in the larger slice area is expected to be toward minimizing IC parts required to implement a system, while maintaining the advantages of microprogramming and flexible architectures. The use of a single 8 or 16-bit RALU to create a microprogrammed processor of the same word length will be of particular impact along these lines. The bit-slice microprocessors will continue to be popular to those applications where speed and flexibility are areas of primary concern. These areas include the use of bit-slice for custom controller design, processor design or emulation, for implementation of special instructions to augment other processing, and for special processor (e.g. FFT, array) development.

4.3.1.3 Special Purpose Processors

Special purpose processors refer to a single chip implementation of a special digital processing function or set of related functions. In contrast to general purpose monolith or bit-slce elements, these special purpose chips are characterized as being extremely application oriented with at most limited programmability. Special processor chips tend to make feasible certain microprocessor-based systems which would not be feasible otherwise. In other words, a special processor/microprocessor team can make possible an application for which an unassisted microprocessor could not deliver adequate performance. Alternatively, a single microprocessor/special processor chip combination can do a job which would otherwise require a multiplicity of microprocessors. Although the BIU functions (e.g. MIL-STD-1553) fall into the special purpose processing category, such devices are discussed separately due to their particular significance to this study. Existing or probable special purpose processing chips are classified into the following categories:

- o Digital Signal Processing
- o Communications and I/O Processing
- o Special Arithmetic Processing
- o Device Controllers
- o Memory Management
- o Cryptography Functions
- o Software Oriented Processors

A market for special purpose processing chips exists because many microprocessor-based systems require additional functions which cannot be effectively performed by the microprocessor chip itself. These functions are typically not found on the microprocessor chip for at least one of the following reasons:

1) The special function is complex, requiring many logic gates and is therefore not technically feasible within the current level of integration.

2) The total chip pinout problem is aggravated by the nature of the function, which would require the use of an undesirably large number of pins on the package. This reason excludes many functions which are both technically feasible and sufficiently marketable.

3) The function does not satisfy a sufficiently large market to justify its inclusion onto the microprocessor chip itself, but it does satisfy a sufficient market to warrant its implementation on a single LSI chip.

It can be expected that future special purpose processors will continue to follow this trend, with migration onto the microprocessor chip only when the previous criteria are no longer of issue. New special purpose processing elements will evolve from functions which are not presently widely used or have not yet been standardized to a level to justify an LSI chip development.

4.3.1.4 Microprocessor Costs

The approach used to derive microprocessor cost estimates was to compare past cost performance against present costs to predict the 1982 microprocessor cost position. As market volume increases because of users familiar with a product, the prices historically drop. Competition by multiple manufacturers also strongly impact cost. These pressures have in the past dominated microprocessor costs and driven the overall costs of established devices downward.

Two applicable cost models include a general 30% per year reduction in cost and an exponential reduction in cost to maturity. Figure 4-2 shows these curves as applied to both the monolithic MOS and bipolar bit slice device prices.

The industry leaders (e.g. 8080 and 2901 of recent years) prices will follow the curves of figure 4-2. A reasonable target floor price for established MOS processors for the commercial market is $8 in quantities of 100. The 8086 and Z8000 type processing units introduced in 1978 and 1979 will be representatives of this class. Higher performance and multiple-microprocessor units will appear in 1982 at costs of $150-200, an estimated 30% greater than top-of-the-line units in 1978. Similarly, bi-polar bit slice device price estimates are targeted at $25 in 1982. These units will be performance-improved versions similar to the AM2903 processor of 1978.

Costs for military environment devices will continue to be significantly greater than that for commercial devices. Military grade MOS (or equivalent) microprocessors will cost $20-30 in 100 unit quantities. Bit slice devices will cost $90-100 in 100 unit quantities. These estimates are based on 1978 military-commercial cost differentials and the 1982 commercial price projection. This estimate is considered conservative from the viewpoint that military device costs will further escalate should these devices represent a smaller fraction of the microprocessor market. As a final cost factor, inflation will escalate costs across the board, with an example annual 5-6% rate yielding a 30% delta to be added to projected 1982 and 1983 costs.

4.3.2 Memory Technologies

Semiconductor memory products are challenging the territory once exclusively held by magnetic core memories. The products include a wide range of volatile and nonvolatile memory products. Advances in speed, power and density in core memories have slackened considerably. This is forcing the application of core to move from the standard main memory to the area of fast mass storage. Semiconductor devices continue to double in density each year. Table 4-4 shows those components, both commercial and Mil-qualified, which are expected to be available in 1982. Should nuclear hardness be a requirement, the choice of component would be limited to Mil-spec parts using bipolar or CMOS/SOS processes.

Mask-programmed read-only memory (ROM) is programmed from a customer supplied pattern by the manufacturer during the final fabrication step. Due to the masking operation required for ROM's, its application is limited to volume orders. Currently, use of ROM's are more economical than EPROM's in volumes of 100 or more and more economical than PROM's in quantities of 1000 or more. As ROM's move above the 64K bit threshold, the standard 24 pin configuration is inadequate forcing a move to larger pin-outs (probably 28 pin).

The PROM market is currently dominated by bipolar technologies which offer high performance. Future trends predict high density MOS PROM's becoming popular. Once reliability questions are resolved, CMOS will make strong gains into the bipolar realm. Recent developments include a NMOS PROM which is in reality an EPROM packaged in opaque plastic. Although demand for bipolar PROM's now exceeds that for MOS EPROM's, it is expected that by next year (1979) the EPROM sales will exceed those for PROM's. This trend is expected to continue, due to the

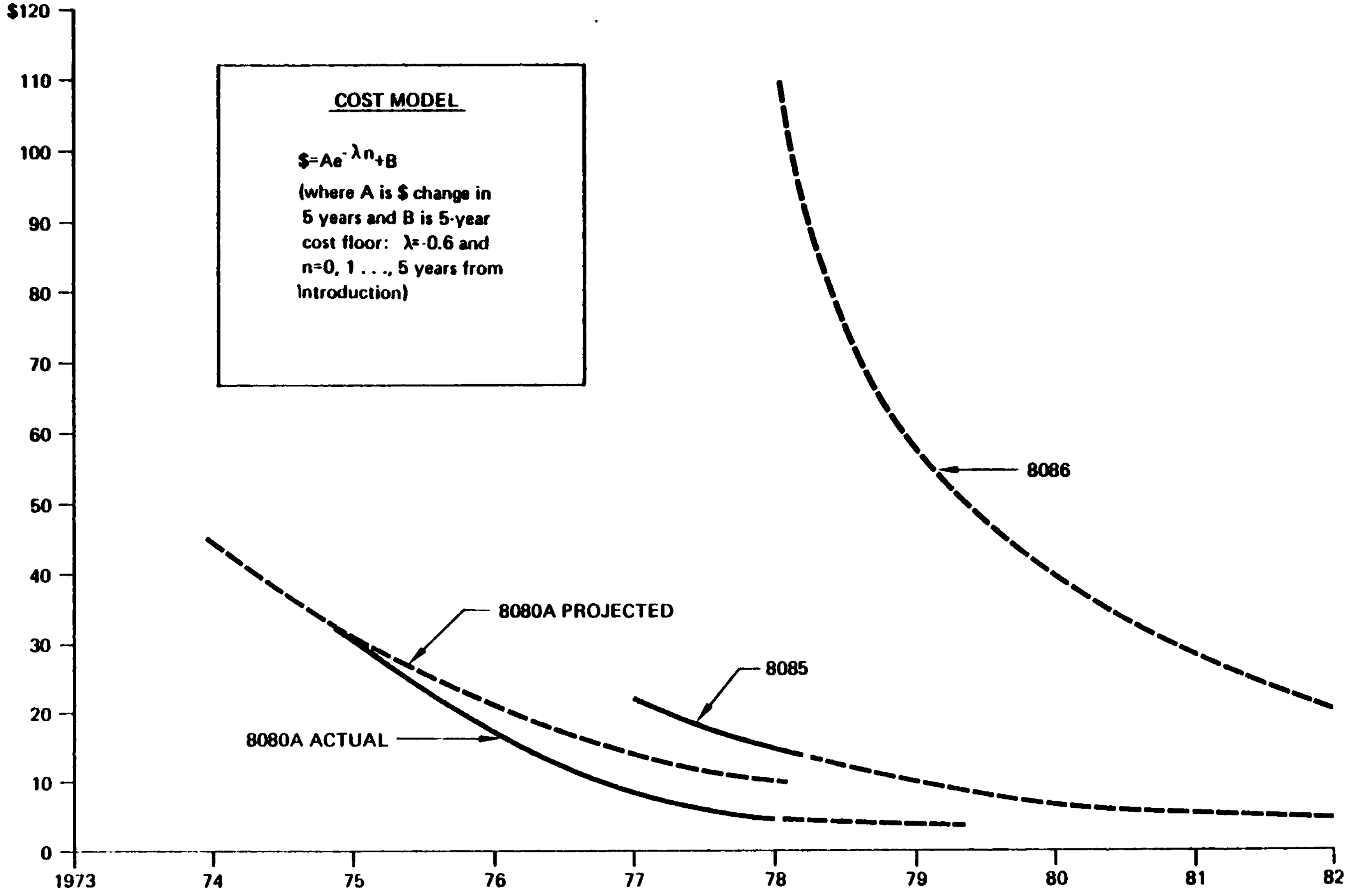


Figure 4-2. Microprocessor Costs

Table 4-3 PROJECTED 1982 MEMORY COMPONENTS

Device Type	Commercial Parts (# Bits/Process)	MIL-Spec Parts (# Bits/Process)
ROM	1M/NMOS	256K/NMOS
PROM	256K/NMOS 32K/Bipolar 16K/CMOS	64K/NMOS 16K/Bipolar 8K/CMOS
EPROM	256K/NMOS 16K/CMOS	64K/NMOS 8K/CMOS
EAROM	16K/MNOS	4K/MNOS
Static RAM	64K/NMOS 16K/CMOS	16K/NMOS 8K/CMOS
Dynamic RAM	256K/NMOS 64K/Bipolar 16K/CMOS	64K/NMOS 16K/Bipolar 4K//CMOS
Magnetic Bubble	1M	-
CCD	1M	256K

Note: NMOS includes special scaled processes such as HMOS, VMOS, etc. Bipolar refers to TTL, ST^{2}L, I^{2}L and I^{3}L technologies CMOS includes all CMOS processes including CMOS/SOS.

large number of low-volume microprocessor applications. This larger volume will also drive EPROM prices down. EPROM devices are benefitting from the advanced MOS process developments and from continuing success with CMOS. These technologies will increase performance and lower dynamic RAM's into the background. Low power CMOS is expected to move into the dynamic realm, although still at a higher cost.

Magnetic bubble memories (MBM) provide the means for nonvolatile storage through a serial organization of localized magnetic fields. Charge-couple devices (CCD) provide a similar organization with a circulating system of minority carriers. Both memories are implemented using multiples of these natural shift registers on the same chip. Although MBM's have the advantage in being nonvolatile where CCD's are volatile, CCD memories have superior access times, though not into the MOS ranges. Both have potential mass storage application due to high packing densities and low cost. Bubble memories have problems at high temperatures which will restrict their military applications.

4.3.3 BIU Technology

The projection for the avionic BIU is based on continued use of a MIL-STD-1553 class of avionic buses or similar military standard. This BIU is characterized by a bi-directional, single-thread serial bus with high bandwidth (1 Mb/s), relatively long length (up to 300 ft) and reasonable multi-tap capabilities (up to 32 taps for the wire bus). This technology forecast assumes a trend that contrasts the microprocessor and memory trends, where an evolution of military parts was derived from the high volume commercial market. Several observations are made to substantiate the anticipated military BIU trend:

1) A precedent for an avionic bus is already firmly established with MIL-STD-1553.

2) No commercial bus standard (defacto or otherwise) is currently available which is reasonably comparable to MIL-STD-1553. Commercial schemes are typically multiple-wire, limited in bandwidth or distance, or have a high overhead associated with transfers.

3) The military applications are historically more aware of fault tolerant considerations than commercial endeavors; therefore, more sophisticated bus protocol features are required. This factor alone could make a new commercial standard unacceptable for avionic buses, even if it met all other requirements for bandwidth, length and efficiency.

4) A two chip LSI implementation of MIL-STD-1553B is available.

The anticipated military origin of a 1982 avionic BIU has one important implication: an initially low volume market which will result in a relatively high parts cost. This does not preclude eventual commercial acceptance of the military standard which would drive chip prices down. Therefore, commercial use of the military BIU is desirable. The two chip MIL-STD-1553B BIU is expected to be available in 1980 at a total cost of $400-500. Following entry into the market by these chips, the subsequent sales volume is critical to its cost future. With at least some commercial use, lower costs are expected. Two factors have the potential for dramatically lowering the cost:

1) Entry of a competitive second source for the BIU chips.

2) Development of a single chip BIU.

The effect of these is shown in figure 4-3. This figure illustrates general effects and is not intended to predict the dates of such occurrences. Should a step in the evolution not be achieved, the cost curve will continue along the same line, eventually leveling off, and possibly rising as sales diminish.

Associated with the evolution of the BIU, several significant factors are mentioned which have the potential for drastically changing the conclusions drawn to this point. Although these are not considered to be highly probable events, they deserve consideration, if for no other reason than to realize the uncertainty in the BIU technology forecast due to a lack of precedent data.

1) A commercial bus standard will be adopted by the military, which differs from MIL-STD-1553 trends.

2) A comparable serial bus standard will evolve from commercial applications which is acceptable to the military.

3) A backbus standard will be adopted and become popular.

4) Commercial microprocessor manufacturers will develop BIU chips to interface with the military standard bus and be compatible with their respective processor bus architectures.

Although the four conditions discussed above are not considered highly probable, any one would have a dramatic effect on the future of the BIU.

4.3.4 Environmental Considerations

Environmental considerations cause the military requirements to differ from the normal ambient environment required by many commercial applications. The ability of components to meet these environment requirements for avionic applications, with the exception of nuclear environment, are normally assured by having the parts screened to Class "B" MIL-STD-883. The extent to which memory devices may be Mil-qualified is largely dependent on the technology used.

Manufacturers and users of semiconductor devices recognize that there are three significant causes for reduced reliability: mechanical integrity, the thermal compatibility of materials, and the quality of the final seal on the device. The reliability problem is significantly reduced by defining and monitoring manufacturing procedures and 100% environmental testing of the devices to identify and eliminate the potentially weak devices. In essence, MIL-STD-883 is a level of confidence on the reliability of the part over the environmental range.

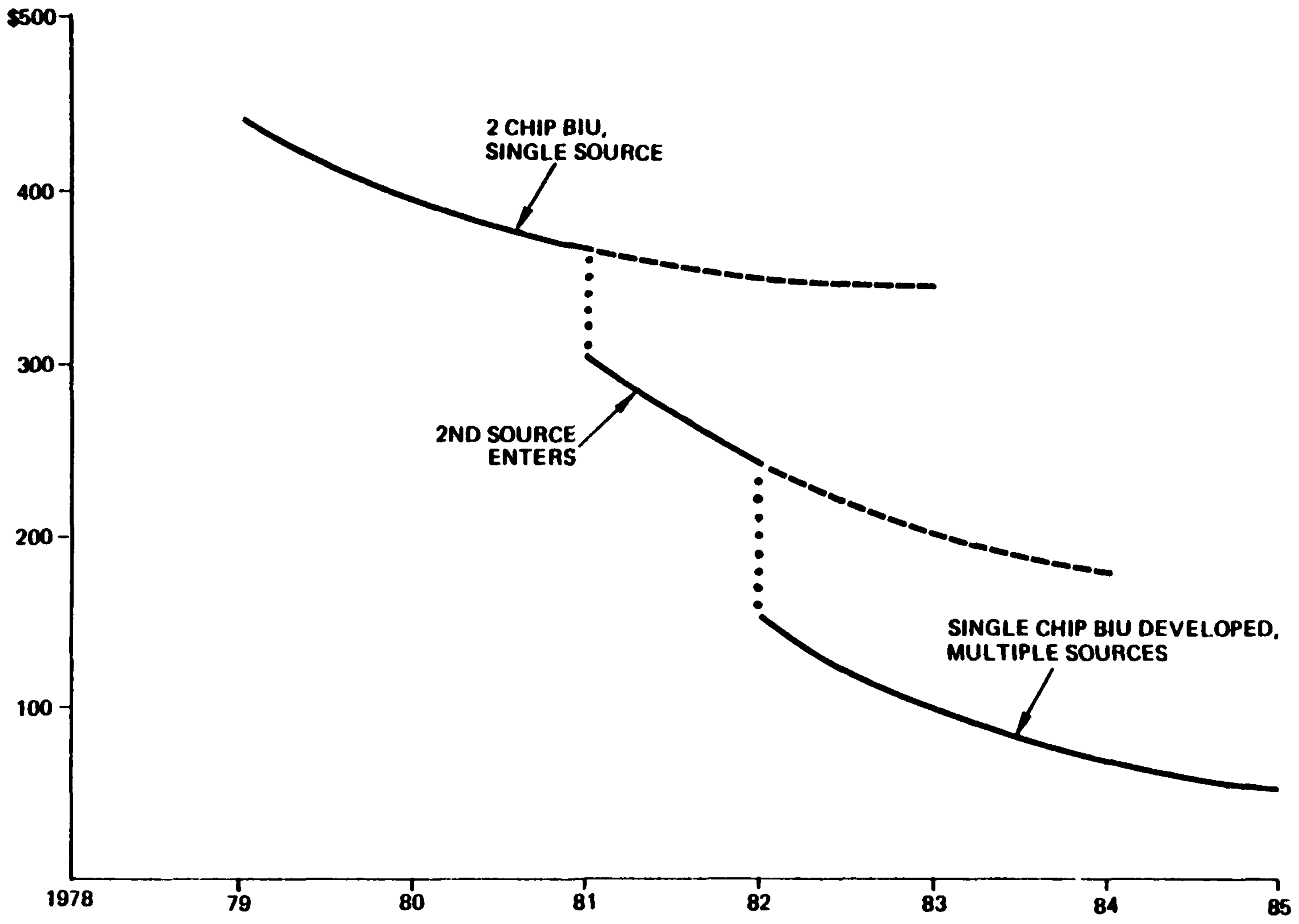


Figure 4-3. BIU Cost Effects

The electrical tests required to exercise the programmable devices are extensive and complicated. The test requirements are dependent on the microprocessor or device functional architecture. Each different device requires considerable time and expense to develop its test procedure. The Qualified Parts List is projected to have a very limited number (one or two) of 16-bit microprocessors in the 1982 time frame. The same is true of memory devices.

At present, nuclear requirements are specified in terms of total levels and dose rate levels determined for the specific application. Some technologies are "harder" than others and some technologies can be made harder by special manufacturing considerations. Table 4-4 summarizes the approximate nuclear hardness of the "off-the-shelf" processor technologies. Each technology is assessed against the total dose, neutron fluence and transient nuclear environments. The use of an NMOS processor is severely limited due to the relatively low total dose hardness. The CCD memories exhibit the same limited nuclear hardness as NMOS because of similar technology. The CMOS/SOS technology offers a clear advantage for applications where the total dose requirements are relatively low. Since this technology is dielectrically isolated, it has transient radiation advantages. The three bipolar technologies (I^2L, TTL, ECL) are best suited for applications requiring moderately high levels of total dose hardness or where a large safety factor is applied to the total dose specification to minimize hardness assurance consideration during production. In summary, a clear choice of technologies cannot be made until the specific nuclear specifications, over test levels, and system requirements are known. At that point, a choice can be made and nuclear guidelines issued to the designers to minimize the nuclear effects and transient behavior on the system.

4.4 STANDARDS

4.4.1 Hardware Standards

4.4.1.1 Purpose of Hardware Standards Study

Distributed processing systems naturally require modular software and hardware which can be added or removed. This modular concept when properly applied, has been shown by numerous studies to generate lower life cycle costs for systems involved in such reconfigurations and retrofits. These studies also indicate that these modules must be carefully planned around system level, technology independent software and hardware standards.

The military is the only sector that, because of its organization structure and a commonality of system level problems, can actually derive and invoke top down standards. In contrast, several commercial standards evolve from:

1) A planned expansion of a family to accommodate a range of requirements (e.g., PDP-11 series, IBM 360/370 series)
2) Building block family (e.g., TTL series)
3) A compromise between providing the latest LSI and VLSI technology
4) Solving problems in fabrication limitations (e.g., decisions on the architecture of many microcomputers).

Table 4-4 APPROXIMATE NUCLEAR HARDNESS FOR THE PROCESSOR MANUFACTURING TECHNOLOGIES

	NMOS					
	Dynamic	Static	CMOS/SOS	I^2L	T^2L	ECL
Total Dose RADS (Si)	1-2,000	1-2,000	3-8,000	10^4	10^5	10^5
Neutron Fluence (n/cm^2)	10^{13}	10^{13}	10^{13}	10^{12}	10^{13}	10^{13}
Dose Rate Upset Level (Rads/sec)	10^7	10^8	$5x10^9$	10^8	10^8	$2x10^8$
Dose Rate Latch-Up Level (Rads/sec)	$10^8 - 10^9$	$10^8 - 10^9$	Immune	1	$10^8 - 10^9$	$10^8 - 10^9$

1 Latch-up is possible with this technology - has not been observed to date.

Some commercial standards are derived from a top level engineering designs (i.e., IEEE Standards, EIA Standards, etc.). In general, the commercial products are geared toward increasing a manufacturer's profit by satisfying the general public's requirements and quickly captivating a volatile market by being the first with a special feature. Subsequently, some standardization is adhered to in order to effectively support these products. The military standards are normally intended to supplement the commercial standards where voids occur (i.e., MIL-STD-1553, MIL-STD-883, etc.).

4.4.1.2 Study Approach

The military is often trapped when improved technology obsoletes a standard hardware approach. A sound standard must encourage, not restrict, design innovation. While there are many hardware standard thrusts in motion, standards which are written and organized as a functional hierarchy and which take into account the application are considered the most useful. As a result, the need exists for an overall hardware standards metastructure whose organization considers the various standards levels as they apply to a typical avionic applications (see figure 4-4). Such a hierarchical organization considers the standards problem from a top-down approach, considering first global network interfaces and moving downward to more precise specifications. In such a metastructure the core and center rings are admittedly design restrictive, with technology independence increasing as one moves away from the core. The lowest standard which may be considered reasonably independent from technology is that of functional module standards. This gives a three level hierarchy of "global bus/backbus/functional module which should be applied to the avionics environment.

4.4.1.3 Hardware Standards Studied

In this study it is believed that standardization at the sensor/LRU level would greatly simplify the aircraft integration task; however, to limit the standardization effort at that level only would be unwise. It is also necessary to define the backbus, microprocessor, memory, special processor and sensor I/O interfaces using existing standards whenever possible. This definition is necessary to encourage either military programs or component manufacturers to produce products to these standards. The level of interfaces and module standards deemed essential for a typical AASMMA application are reflected in figure 4-5. In essence, the typical module partitioning serves to identify the standards. The partitioning approach was heavily influenced by existing standards, mainly those used on the DAIS program. The one noticeable void in the DAIS standards was that of an acceptable "backbus" standard. Standards of one form or another exist for the other modules.

The network bus standard is expected to continue along MIL-STD-1553 trends due to the firm precedent already established by this standard. The commercial arena lacks a comparable serial bus standard and it is doubtful that one which would be acceptable for avionic applications will develop to be a serious alternative to 1553 in the early 1980s.

A subsystem or "backbus" interface standard is considered a firm requirement for the orderly development of future distributed systems. This will most likely be a parallel bus which will interconnect the microprocessor, memory, special devices and any sensor I/O ports. A backbus standard would require a full functional and electrical specification of the interface and would address in detail the issues of memory access, I/O, DMA transfers and hardware interrupts.

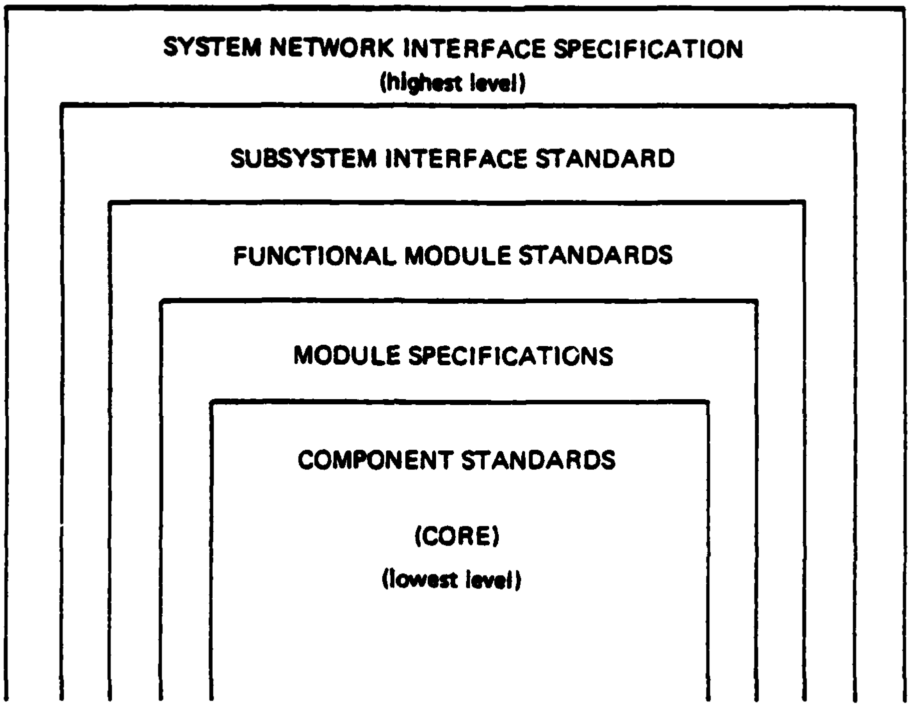

SYSTEM NETWORK INTERFACE SPECIFICATION
(highest level)
SUBSYSTEM INTERFACE STANDARD
FUNCTIONAL MODULE STANDARDS
MODULE SPECIFICATIONS
COMPONENT STANDARDS
(CORE)
(lowest level)

Figure 4-4. Hardware Standards Metastructure

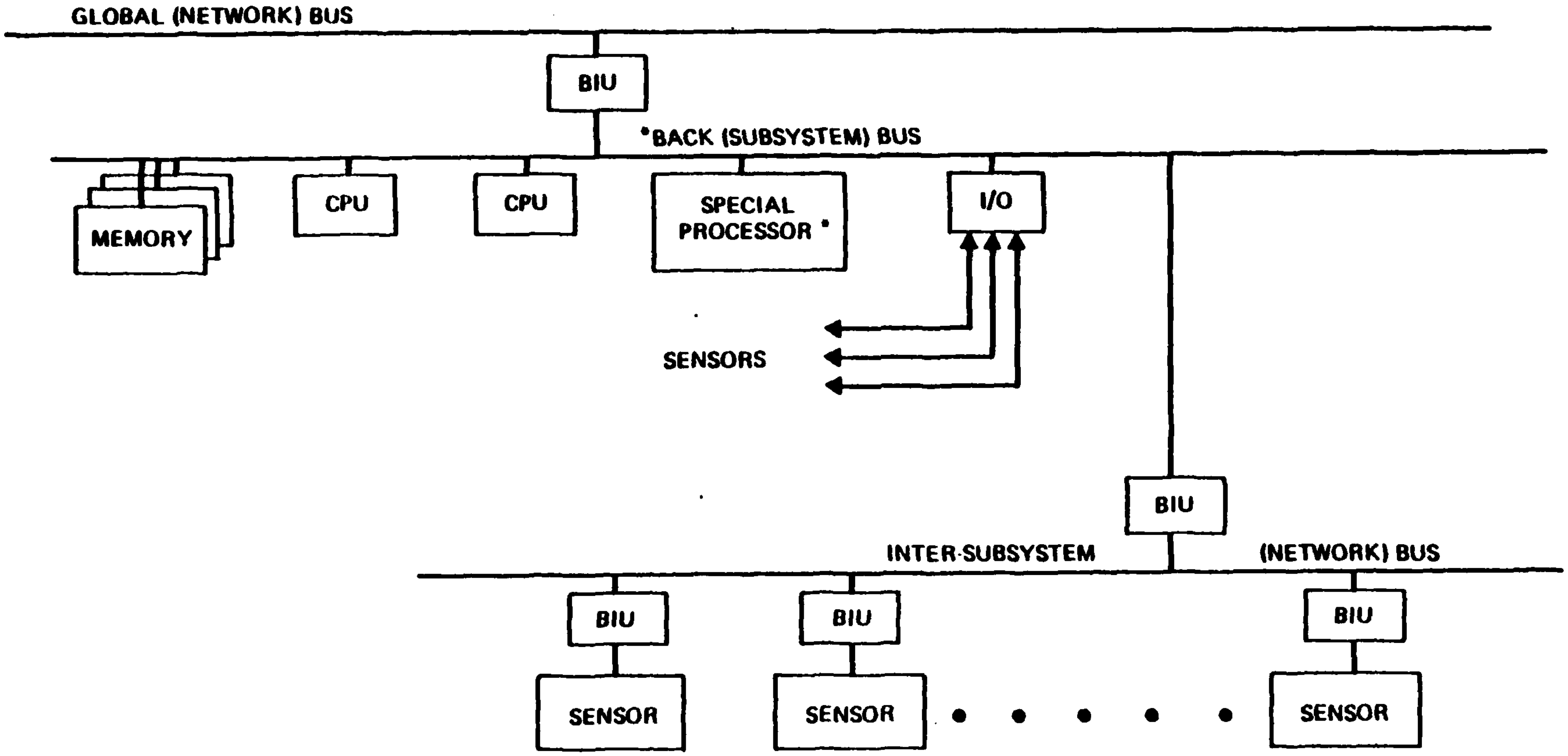
While such a specification obviously constrains many system aspects, a well chosen standard which looks toward future developments, such as multiple processors and shared resources on the backbus, should be sufficiently long-lived to be worthwhile.

Since the subsystem bus will be utilized as a convenient rather than a performance optimized interface, a family of backbus standards may develop rather than a single one. This family will most likely be partitioned into "sensor-preferred" bus standards (e.g., IEEE 488 instrumentation bus) and "microprocessor-preferred" standards (e.g., Intel Multibus, ProLog Std Bus, National Microbus, IEEE S-100 bus, etc.), with the choice of standards based upon the individual requirements of the subsystem. This approach will give sensor manufacturers flexibility to choose a cost effective method for producing sensors with embedded processors while allowing the integration contractor to limit the proliferation of interface mechanisms.

No acceptable military standard was found for the backbus application. The Navy's SEM program did establish the requirement for a card interconnect standard. There are several commercial bus standards (defacto or otherwise) for both sensors and microprocessors buses which appear to be acceptable for avionic backbuses. Choosing a standard from these established commercial buses would have the potential for lower component costs.

The next level of standard hierarchy is the functional specification of modules which may interface each of the subsystem bus standards. For a microprocessor module, functional specification would include: instruction set, interrupt handling, I/O and error handling details. Candidates for such specification not only include a MIL-STD-1750 instruction set, but also popular defacto commercial standards (e.g., LSI-11, 8086, etc.). A key question identified during the technology forecast was the potential absence of a MIl-STD-1750 monolithic microprocessor. The new commercial 16-bit monoliths which resemble mini-computers in features and potential performance appear to be applicable for avionic processing. Since future cost effectiveness of embedded systems will be obtained using monolithic processors, the computer family and microprogrammed building block philosophy may not provide as great an advantage as earlier anticipated.

Other functional modules that must be considered are memory and sensor I/O. Memory modules will be functionally specified by a backbus standard. Sensor I/O interfaces include parallel digital, serial digital, analog, synchro and usually pulse trains. While it is felt that a set of definitions (such as the DAIS interface modules) is possible, it is likely that the I/O interface will be built into the sensor. Consequently, the definition of the backbus will suffice for defining all sensor I/O. A special case of the I/O interface is the BIU, forming the data path between a bus network and the backbus. The functional module is defined through specification of the network and subsystem bus standards. Special processing modules like floating point, FFT, matrix manipulation, sine and cosine, and high precision arithmetic must also be considered. While it is possible to include these functions in the functional CPU module, this would result in a revised processor module standard. Alternately, such functional modules could be integrated and accessed by the processing element via the backbus. Since technological trends will result in both approaches being used,



*(e.g., FFT, FP, Sin/Cos, etc.)

Figure 4-5. Example AASMMA Configuration

both must be considered during any standardization effort. Furthermore, the standards definition must consider the fact that these functional modules could be implemented strictly in software within the processor, by a dedicated CPU, or by using special hardware logic.

The lower levels of the organization require precise hardware specification of the defined functional modules and detailing to the component level. These levels are considered too technologically dependent to be applied on a large scale basis.

4.4.1.4 Summary of Hardware Standards Study Results

The study of avionic microprocessor hardware standardization considered four candidate areas: 1) the high speed serial global bus of the MIL-STD-1553 class; 2) the parallel subsystem bus (backbus) for interfacing sensors and microprocessor hardware; 3) subsystem architectures, specifically for microprocessors, memory, I/O and special processing; and finally, 4) specific hardware module specifications. A summary table of the findings and recommendations is shown in Figure 4-6.

It has long been recognized that global bus standardization is required for successful integration to a multiplexed avionic system. The military is unquestionably dominant in this area, with no applicable commercial efforts found. Being adequate for near future avionic requirements, the well established MIL-STD-1553B should continue to be maintained. However, considering the 10-year development cycle for 1553, the next generation speed-independent bus standard should be investigated now.

Just as global bus standardization is required for multiplex system integration, subsystem bus standards are necessary for distributed avionic system configuration. As far as leadership in the area, the opposite condition exists in the area of subsystem buses. Here the commercial world is especially active, with few standardization efforts ongoing in the military. Considering the cost of standards development and possible hardware payoff in the use of existing LSI support chips, the commercial bus standards should be fully exploited by the military. Specific consideration should be given to the IEEE 488 bus for "sensor-preferred" applications and the Multibus for "microprocessor-preferred" subsystems. The existence of available support hardware will make compliance with such standards cost-effective for the sensor manufacturer.

Subsystem architecture standardization results in significant cost benefits (particularly software) without technology dependence, although studies have shown the relative cost saving of one architecture over another are small. This area of standardization is likewise beneficial to the system integrator and thereby cost-effective for the Air Force. The commercial marketplace contributes many candidate defacto standard architectures which appear applicable for avionics.

STANDARDIZATION APPROACHES	SYSTEM NETWORK BUS STANDARD	SUBSYSTEM BUS STANDARD	SUBSYSTEM ARCHITECTURE STANDARDS	HARDWARE MODULE SPECIFICATIONS
FINDINGS	o NETWORK BUS STANDARD ESSENTIAL TO AVIONIC MULTIPLEX SYSTEMS o MIL-STD-1553 HAS TAKEN 10 YEARS TO BECOME WELL ESTABLISHED o MIL-STD-1553B IS BOUND WITH 1 MB BANDWIDTH AND IS NOT OPTIMAL FOR DISTRIBUTED SYSTEMS o COMMERCIAL WORLD HAS NO BUS STANDARD COMPARABLE TO MIL-STD-1553B	o SUBSYSTEM BUS STANDARD ESSENTIAL TO DISTRIBUTED AVIONIC SYSTEMS o TWO AREAS OF BUSES IDENTIFIED: "SENSOR-PREFERRED" "MICROPROCESSOR PREFERRED" o MANY APPLICABLE COMMERCIAL BUSES EXIST, MANY WITH LSI SUPPORT CHIPS o MILITARY BUS STANDARDS HAVE NOT CONSIDERED MULTIPROCESSOR BUS PROBLEM	o ARCHITECTURE STANDARDIZATION HAS MANY ADVANTAGES OVER EXACT HARDWARE SPECIFICATION o APPLICABLE COMMERCIAL DEFACTO STANDARDS EXIST FOR PROCESSORS, MEMORY AND I/O o MONOLITH MIL-STD-1750 MACHINE WOULD BE EXPENSIVE TO DEVELOP, BUT BIT-SLICE MACHINE HAS HIGHER LIFE-CYCLE COSTS o MIL-STD-1750 IS DEFICIENT IN MANY AREAS WHEN COMPARED TO 8086, Z-8000 AND MC 68000 o ARMY/NAVY MCF STUDY (WHICH CHOSE PDP-11 ARCHITECTURE) ESTABLISHED AND EVALUATION APPROACH o COST ANALYSIS HAS REVEALED DIFFERENCES BETWEEN ARCHITECTURE STANDARDS IR PROPORTIONALLY SMALL o SPECIAL PROCESSING AREA MOST ILL-DEFINED	o EXACT HARDWARE SPECIFICATION STANDARDS ARE RESTRICTIVE TO DESIGN AND TECHNOLOGICAL PROGRESS o STANDARDS IN THIS AREA HAVE NOT BEEN PARTICULARLY SUCCESSFUL AND OFTEN SHORT-LIVED
IMPACTS	o NETWORK BUS STANDARDIZATION HAS WORKED WELL FOR THE INTEGRATOR AND SENSOR MANUFACTURERS. IT HAS RESULTED IN COST SAVINGS TO THE AIR FORCE	o ALTHOUTH REQUIRED BY INTEGRATOR FOR DISTRIBUTED AVIONICS, SUBSYSTEM BUS STANDARDIZATION WILL NEED TO BE ADVANTAGEOUS TO THE SENSOR MANUFACTURER TO BE ACCEPTABLE TO HIM o AIR FORCE SAVINGS WILL BE REALIZED AS A RESULT OF SUCCESSFUL DISTRIBUTED AVIONICS	o ARCHITECTURE STANDARDIZATION IS ADVANTAGEOUS TO THE INTEGRATOR, BUT DESIGN RESTRICTIVE TO THE SENSOR MANUFACTURER o ARCHITECTURE STANDARDIZATION CAN LEAD TO SIGNIFICANT SOFTWARE SAVINGS FOR THE INTEGRATOR AND AIR FORCE	o HARDWARE SPECIFICATION STANDARDS ARE DESIGN RESTRICTIVE TO ALL PARTIES
RECOMMENDATIONS	o THE AIR FORCE SHOULD CONTINUE ITS LEADERSHIP IN THIS AREA WITH CONTINUED MAINTENANCE OF MIL-STD-1553 o THE AIR FORCE SHOULD START NOW TO DEVELOP THE NEXT GENERATION SPEED-INDEPENDENT NETWORK BUS STANDARD	o THE AIR FORCE SHOULD UTILIZE COMMERCIAL SUBSYSTEM BUS STANDARDS WHEREVER POSSIBLE	o THE AIR FORCE SHOULD USE STANDARD COMMERCIAL ARCHITECTURES WHEREVER POSSIBLE o MIL-STD-1750 SHOULD BE EVALUATED AGAINST THE "BIG THREE" USING THE MCF METHODOLOGY o THE VHSIC (AND SIMILAR PROJECTS) SHOULD CONCENTRATE ON FILLING THE VOIDS	o STANDARDS AT SUCH A LOW LEVEL SHOULD BE AVOIDED o SOME AMOUNT OF STANDARDIZATION CAN BE ENCOURAGED BY THE AIR FORCE

FIGURE 4-6 Standardization Approaches For Avionic Microprocessor Hardware Summary Table

Sufficient evidence exists to warrant evaluation of the 8086, Z-8000 and MC 68000 architectures against the Air Force MIL-STD-1750 processor architecture in order to decide the course of future avionic systems based on performance and cost measures. Elsewhere commercial standards should be utilized wherever possible with programs such as the VHSIC "filling in the voids," particularly in the area of special processing.

hardware module specification standards both at the chip and board levels, are highly design restrictive and historically short-lived. Although cost savings are perhaps more apparent at this level, they are proportionally small. Some amount of standardization, however, can be encouraged by the military with "preferred" or particularly cost-effective parts being made available.

The average wait time for high priority messages was calculated using the individual characteristics of each information transfer system. The stationary master case assumed that the device requesting the high priority message transmission was a remote processing element. Communication with the remote processing element (for this analysis) would occur on a cyclic basis with each processing element communicating with the master in turn according to a predefined message mix. This assumption, given the message mix and the time required to communicate asynchronous requests, resulted in the stationary master representation in figure 3-1. The nonstationary master was further complicated by the fact that a controller communicating with a device requesting a priority transmission might not be the controller containing the proper information for the message transmission to occur. Consequently, several message timings are shown: one for the proper remote processing element to communicate with the proper controller, one for the processing element containing its own transmission request information, and one for extensive polling which would decrease the response time at the cost of additional overhead. The contention communication assumed that one high priority message was required to be sent and the remainder of the processing elements had lower priority messages to transmit. Each analysis is detailed in the First Interim Report, appendices A, B, C and D.

4.4.2 Software Standards

4.4.2.1 Purpose of Software Standards Study

The goal of any standard imposed on an avionic system is to reduce cost. Reduction of cost may be realized during acquisition or over the life of the system. Cost savings may not be realized when using a standard, except when considered over the life of several avionic systems in the military inventory.

Both language standards and software modularity development standards have been shown to be cost-effective for specific systems. The cost benefits of the various standards are known to differ greatly, especially when applied to different types of processing systems. Because most existing language and module development standards were developed for miniprocessor systems, one of the AASMMA study goals is to determine the applicability and cost benefits of the standards when applied to microprocessor based, distributed processing systems.

4.4.2.2 Study Approach

Based upon a review of software literature (performed in Task II, and recorded in Appendix G of the AASMMA First Interim Report), significant software cost factors were identified. These cost factors are:

Impacted by Software Standards

- o Complexity*
- o Transferability*
- o Schedule*
- o Computer Utilization*
- o Number of Processors
- o Type of Source Language*
- o Operational Life of Program
- o Number of Block Changes
- o Amount of Support Software*
- o Number of Instructions

Independent of Software Standards

- o Instruction Mix
- o Peripheral Devices Used
- o Test Requirements

* Those factors that are directly affected by the various standard approaches.

Using cost models selected and developed in Task II of this program, an analysis was made of the cost factors listed above to determine the cost impact of software standards in microprocessor based, distributed processing avionic systems.

4.4.2.3 Software Standards Studied

Considering the emphasis DoD has placed on language standardization, all forms and combinations of language standards were investigated in this study. These language standards are:

- o Standard HOL (family of HOL's)
- o Standard HOL (single)
- o Assembly Language (AL)
- o Standard AL (standard instruction set)

Also studied was the impact of a structural software development standard such as the one being used by DAIS. The analysis assumed only the use or lack of use of a standard. Specific features of a language were not identified. Similarly, the value of specific aspects of a software development standard (e.g., naming conventions, data transfer techniques) was not analyzed.

4.4.2.4 Summary of Software Standards Study Results

The results of the cost analysis performed on the various forms of software standards are shown in figure 4-6 and are explained in Appendix I of the first interim Report. Due to the uncertainties in the accuracy of the software cost model used in this analysis, a band of costs (shaded area) is given, rather than a single point cost estimates. Even with possible inaccuracies of the cost, the analysis clearly showed that the adoption of DAIS-like software development standards (modular, control independent, application software) woud lower software costs. Further, the analysis showed that the lowest cost approach was to use a combination of the standard HOL, standard instruction set, and the application software development standard.

There is too much uncertainty in the accuracy of the cost model to determine if the adoption of a HOL alone will lower software costs. For the same reason, it cannot be conclusively proved from the analysis, that the adoption of a standardized instruction set alone will lower software cost.

Modular software development can be simplified by the top down software development scheme. However, the task DAIS software standards go one step further. They have established interfaces within the operational flight software which only allow communication through the local executive functions. Software transferability, a most important feature in multimission applications, takes on a different perspective in distributed processors than in centralized processors. It is possible to have each computer in a multiple computer arrangement perform just one function. This approach could require transfer only and the transfer could be accomplished by transferring the computer and software as a unit.

It is necessary to alter the DAIS application software development standard and the executive/application software interface standard for distributed, multilevel architectures. In fact, the distributed processing makes it easier to enforce these standards due to the physical separation and modularity of the processing elements. Only the design of the DAIS executive itself needs to be changed (as described in Appendix E of the First Interim Report).

4.5 COST MODELS

Four cost models have been acquired or developed for estimating the cost of acquiring and maintaining microcomputer based distributed architectures for military avionics. These models are:

1) PRICE 83 - hardware acquisition
2) PRICE L - hardware maintenance and support
3) PRICE S - software acquisition
4) AASMMA Software Cost - software maintenance and support

The RCA PRICE system of models has been selected as the most appropriate for estimating avionic hardware acquisition and maintenance costs, as well as, software development cost. The PRICE models are generalized parametric models that can be calibrated for each user's applications. Being parametric, the models lend themselves to "what if" analysis and sensitivity analysis where small variations in some parameters can be measured. A software maintenance cost model was developed specifically for this study. This model was designed to be sensitive to standardization issues such as HOL versus AL.

The RCA PRICE 83 model has been used with good results for several years to estimate the cost of military hardware. In fact, the model has become the standard hardware acquisition cost model for military programs.

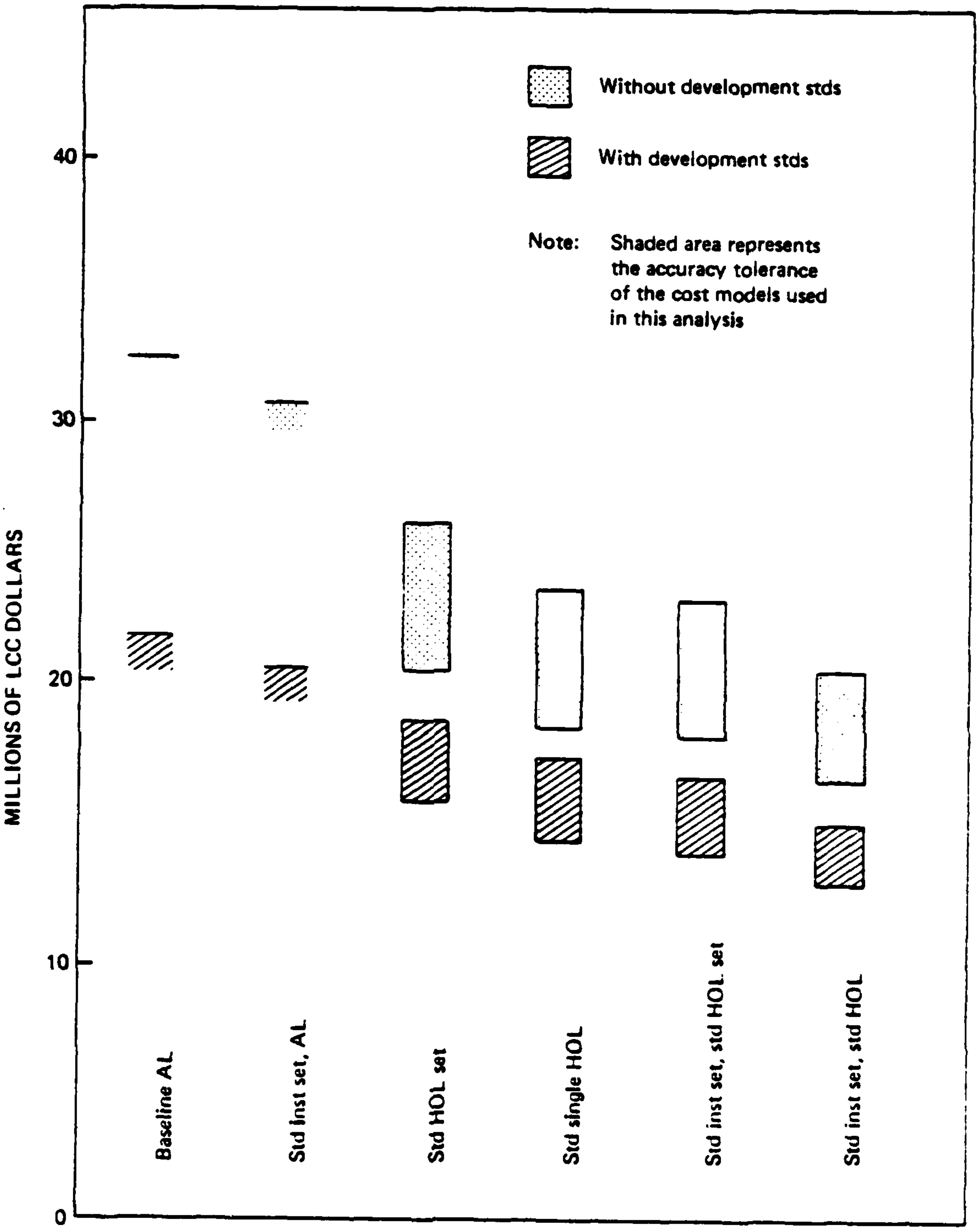


Figure 4-7. Summary of Software Standards Analysis Results

During this study, both the PRICE L and the AFLC LSC model were compared to determine which model was to be used in the study. The analysis of PRICE L and AFLC LSC showed that the cost predictions of the two models track very closely. PRICE L was chosen over AFLC LSC because it is relatively simple to use and because it is relatively easy to perform sensitivity analysis with this model. Another advantage of the PRICE L model was that the outputs of the hardware PRICE model to describe the hardware attributes (e.g., MTBF, MTTR, LRU cost) can be used directly by PRICE L, and it accepts inputs that define the support concept (e.g., number of locations, years in operation).

Software development cost modeling has not advanced to the level of hardware cost modeling. Many models are in use, each of which give widely varying cost estimates. The analysis documented in Appendix G of the First Interim Report showed software cost models to be accurate in some cases and the same model to be in error by over 500% in other cases. Of the models investigated, the PRICE model appeared to be the most consistent cost predictor. The model usually predicted within 10% of actuals and in the worst case, it was in error by a factor 2.7.

Like the hardware models, it is parametric, thus facilitating sensitivity analysis and determination of delta cost for a variety of alternatives. Significant cost parameters are size of program, type of code, reuse of previous design and code, schedule, complexity and peripheral devices used. Outputs from the model include a cost estimate by cost elements (e.g., design, documentation) and sensitivity of the estimate.

Few software LCC cost models are in existence and those which are available do not contain the relationships necessary to evaluate software standards. To quantify the effect of software standards a model was synthesized from existing software LCC models, literature sources and available software LCC data. The significant cost facts that have been incorporated into this model are number of block changes, years of operational life, size of programs, amount of software transferability, number of computer types, acquisition cost of operational and support software, and the type of source language.

The accuracy of this or any other software model cannot be ascertained because of the lack of available data. The AASMMA software LCC model was qualified by comparison with existing LCC models. Figure 4-7 compares the predictions of three parametric software LCC models with that of predictions made by the AASMMA model. These predictions are based on the requirements to satisfy the "strawman" avionic system described in Appendix G of the interim report.

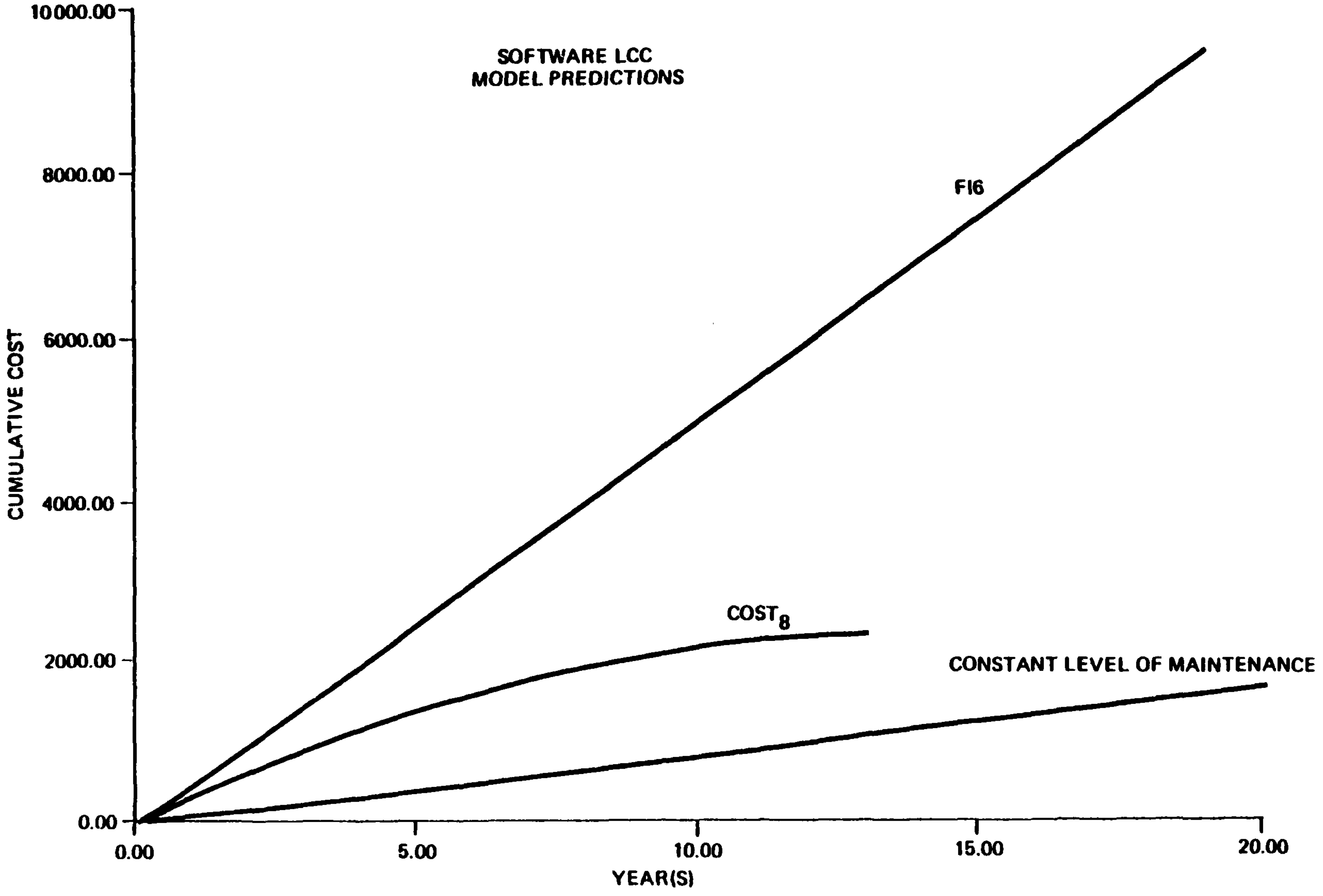


Figure 4.8. Software LCC Model Predictions